QUASI-THREE DIMENSIONAL (Q3D)MODEL OF SNOW AVALANCHE DYNAMICS

by R. I. Perla

October 24, 2017

Multidimensional modeling of snow avalanche dynamics has two objectives. First, to simulate moving avalanche phenomena with emphasis on internal structure. This objective is well beyond the scope of simple, one-dimensional models. Second, to provide a model for consultants to assist their analysis of hazard and effects. In practice, dynamic models are used in combination with field data and statistical models.

Existing dynamic models, used by consultants, set aside simulation of internals, focusing instead on predictions of extreme runout, and estimates of speed, mass and impact pressure in the runout zone. However, as we meet objective one, building increasingly complex 3D models which include internals, we expect as reward for our efforts to better meet the second objective as well.

Internals include mass and speed distributions which separate into flow zones of layers and vortices. These exist within a volume, bounded from below by an erodible slide surface and bounded above by the avalanche-air interface, often not well defined. Internals are observed in the field with great difficulty using penetrating radiation and devices mounted on stands fixed in the path. Internals may also be inferred from destructive impacts on trees and buildings.

Avalanche volumes are typically elongated, snake-like, divided (downstream to upstream) into a head, main body, and tail. The advancing head butts up against a ramp of entrainable snow which is dissagregated into lumps that are devoured by the head and mixed into the moving volume. At the same time some lumps may deposit onto the slide surface, usually from the main-body or tail. Eventually, the avalanche halts, and the entire volume is deposition.

The above phenomena is what a 3D model may attempt to simulate. Clearly, 3D model development requires an enormous effort, combining computer programming and field observations. It's a team effort, open-ended into the foreseeable future.

The meaning of Q3D in the title will be explained shortly, but first some background. The following ideas spring from a series of lectures and exchanges in Prof. Bruce Jamieson's graduate class *"Snow avalanche dynamics and hazard mitigation,"* University of Calgary. As part of the series, in November 2005 we discussed 14 questions for someone who would attempt to model in multi-dimensions. The questions focus mostly on design of inputs and outputs for a multi-dimensional model, less on computational details. The questions seem just as valid today. Let's try to discuss them in order. As we go along, we will explain how Q3D attempts to address the questions (figures and Excel worksheets in the q3d folder will be helpful.)

> *1(a).* Do you believe an important objective of any three dimensional(3D) avalanche model is to predict extreme runout? Or do you believe the model should be calibrated to simulate a runout, which may be determined independently using field and or statistical information?

Discussion. Let's draw on our avalanche path map a centre-line CL extending downslope from the crown line to a maximum runout length L. For large paths, L may be $O(10^3\text{m})$. Most consultants who use and calibrate dynamic models will likely agree that dynamic models can't predict L to better than one percent, or $O(10\text{ m})$, sometimes with errors closer to 10% or $O(10^2\text{m.})$ It may be hoped, and it may turn out, upon introducing more dimensions, internal physics, and structure into the model we will reduce the above errors.

However, if we wish to make progress on the first objective, to simulate internals, we can at least as a start tolerate existing runout errors. We can even go one step further and fix the runout boundary (in general a curved line) and attempt simulations of internals downslope from the crown line, but no further downslope than this cut-off boundary where the simulation terminates and exits. That's essentially what is done in Q3D, where L is considered a user input, or computational cut-off, rather than model output or prediction.

> *1(b).* Is there any application where it is essential that the model predict extreme runout?

Discussion. If the path is unique beyond known comparisons, field data and statistics may be insufficient to estimate extreme runout as required by objective two. In that case, consultants can run a large number of simulations for a range of lower boundary inputs. Some simulations of speed and mass distribution may appear more reasonable than others, and it may be possible to narrow the range of extreme runout. Another application where it would seem crucial a model simulate extreme runout is for a path with a retaining dam in the lower track or runout zone. The better we meet objective one, i.e. simulate internals, the better we can simulate the complex interaction of avalanche and dam.

> *2.* Should the model predict lateral boundaries (avalanche width, which in general varies along the path from start to stop position,) or should the widths be preset by the user, and entered as model inputs?

Discussion. Dynamic models require a user to delineate the entire perimeter of the start-zone. Should the user also preset the path width below the start-zone down the entire path to the lower boundary? The alternative is to allow lateral propagation and entrainment with no preset lateral boundary. In that case, lateral boundaries and total entrainment would be outputs computed by the model. Instead, if we preset the complete perimeter of the path swept by the avalanche volume from crown to lower boundary, then we can also preset total entrained snow at all positions swept by the leading edge from start to stop. Intuitively, this seems the simpler alternative, at least with respect to meeting objective one.

In Q3D, path width is a user input, a side to side map distance, perpendicular to CL. However, Q3D now inherits the problem of how best to constrain motion to within this preset lateral boundary. This will be discussed shortly.

> *3.* Should the model predict flow-heights? Or should the user preset flow-heights down the path?

Discussion. Let's designate flow-height as H, a function of position and time. The short answer is that H is an important output of dynamic models, it is not input. H is the distance from the slide surface up to the ill-defined, avalanche-air interface. It's thickness is roughly $O(1\text{ m})$ to $O(10\text{ m.})$ H is strongly controlled by the sum of start mass plus entrained mass, less deposited mass. H is also controlled by diffusion of granular snow lumps and powder. Q3D models diffusion using Monte Carlo algorithms, essentially random walks. Again, H cannot be preset in dynamic models.

> *4.* The cross-sectional area of avalanche flow is irregular and complicated. What simplified cross-sectional geometries can we introduce?

Discussion. Terrain concavities such as gullies converge avalanche flow and increase H. Terrain convexities such as fans diverge flow and decrease H. In a simplified geometry, cross-sectional flow area is lens-like, bounded by two arcs, a lower arc at the slide-surface and an upper arc at the avalanche-air interface. As just noted, the lower arc can be concave or convex, the upper arc is generally convex. The upper and lower arcs meet at the edges of the line which spans the lateral width.

Q3D makes a further geometrical simplification of the cross-section. It replaces the curved lens-like shape with a rectangle. The width of the rectangle is the lateral width, inputed by the user. The height of the rectangle is the H computed in the simulation. Thus, we have fixed a constant H across the width. Conservation of mass must be accounted for thru the rectangular cross-sectional area. However, internals can only be simulated in the vicinity of CL, away from the side walls. Internals cannot be simulated near the lateral edges and corners of the rectangle.

Thus we end up, not with a full 3D simulation of internal structure, but a quasi-3D simulation which is intended to simulate internals only in the vicinity of CL.

> *5(a).* Avalanche volume can be divided into a large number of small volume elements. A model can output variables for each element. What are the most important output variables which can be associated with each small volume element? Snow velocity? Which velocity components: tangential to the slope, normal to the slope, cross slope (lateral)?

Discussion. It's expected that dynamic models will output for each small volume element the snow velocity component which is parallel to CL. Let's designate that component, which is a function of position and time, as u.

Multi-dimensional simulation of internals also requires computation of the two orthogonal components of velocity, the component normal to the slide surface, and the lateral, cross-slope component. These two components may or may not be included in the output files, but are needed to compute H and mass conservation.

> *5(b).* Snow density in each volume element? If we know speed and density, can we estimate impact pressure? Can you suggest other output variables?

Discussion. It's expected that dynamic models will output density, i.e. the snow mass within a volume element. Models must check for conservation of mass from start to stop, balancing start-zone snow, entrained snow, and deposition which may be distributed over the entire path. Assuming the avalanche volume is a continuous fluid, impact pressure may be estimated as $k\rho u^2$ where k is an empirical coefficient, ρ is snow density, and u is the velocity component parallel to CL.

But we know the avalanche volume isn't continuous, it's a collection of lumps and powder. Impact pressure strongly depends upon impulse forces delivered by lumps. If a model attempts a granular rather than continuous simulation of internals then it may output variables of granular statistics. More on this later.

> *6(a).* An avalanche volume will extend longitudinally from its front (head region) toward its tail. Output variables will have to be displayed to the user along this longitudinal dimension. What is the appropriate spacing of the longitudinal grid for displaying output? 0.1 m? 1 m? 10 m?... Should the longitudinal grid separation be held constant from front to tail? Or to output more details for the head region should the output grid be finer in the head region and coarser in the tail?

Discussion. Numerical computations of avalanche simulation require small time and distance steps. Generally, the smaller the better limited only by computational time. For example, computational steps may be $O(0.1 \text{ m})$ or finer, far too fine to be monitored as output of internals for an event with L as large as $O(10^3\text{m})$.

There are endless ways to build a coarser output grid. Q3D uses a coarse output grid of 100 rectangular prisms which cover the avalanche volume from the advancing front to the tail. The dimensions of each prism will depend on its distance measured upslope from the advancing front. Prism height is the simulated H at that position. The volume of each prism is the product of three orthogonal lengths: H which is $O(1 \text{ to } 10 \text{ m})$ computed in the simulation; lateral path width, inputed by the user, $O(10 \text{ to } 100 \text{ m})$; and a distance ΔS measured parallel to CL as now explained. In order to output greater detail of internals for the head compared to the tail, ΔS of the first prism, i.e. the leading prism at the front, is always fixed at 1.0 meter. For prisms#2 to #100, ΔS increases (stretches) exponentially.

Therefore, as the distance of a prism from the front increases, outputs such as speed and density are averaged over an increasing ΔS.

> *6(b).* Is the avalanche tail usually well defined, or are we forced to display output from the advancing front upslope to the top of the start zone?

Discussion. The avalanche tail is often not well defined. Q3D assumes the upslope tip of the tail is always the top of the start-zone, and assumes that snow may be deposited anywhere on the path from the advancing front to the start zone. The 100 prisms with exponentially increasing ΔS always cover the swept distance from front to start-zone.

> *7(a).* In addition to its longitudinal dimension, the avalanche volume extends laterally across its width, and vertically from slide-surface to maximum flow height. How many lateral grid elements should be outputted to the user?

Discussion. In Q3D, each rectangular prism spans the entire, preset lateral width. We could attempt to simulate more realistic cross-sections which allow H to vary. For example, we could extend Q3D by sub-dividing prisms into columns, and allow H to vary laterally from column to column.

Q3D sets ΔS at 1.0 m for prism#1. If we also set the column width ΔW in prism#1 to be 1.0 m, we will need $O(100)$ columns to fill the lateral width of prism#1. Since ΔS of prisms#2 to #100 increases exponentially, we could also increase ΔW exponentially, substantially reducing the total number of lateral columns. Using that reduction, we may end up with $O(10^3)$ output columns for 100 output prisms.

> *7(b).* As a major simplification could we output dynamic variables only for a rectangular *control volume* of unit width and variable height which follows the avalanche centre-line along its longitudinal dimension?

Discussion. For cross-sections simulated with columns, dynamic variables ρ, and u can be computed and averaged within a rectangular, narrow, 1.0 meter laterally wide *control volume* ($H \times \Delta S \times 1.0$) which straddles CL. For a cross-section with a convex avalanche-air interface, we expect maximum H to occur near CL. As a major simplification, we could output ρ and u only for that control volume.

> *8.* The advancing avalanche entrains snow into its head. Some mass is transported back into the main body of the avalanche and into the tail zone where deposition is possible even while the head is entraining. At the same time there is forward momentum transfer from the main body of the avalanche into the head. This produces intense shear zones, flow separation, and a spectrum of vortices. What is the minimum size of the output vertical grid (the number of vertical elements outputted to the user) so that the user can confirm that the model does in fact simulate flow separation and vortices?

Discussion. Perpendicular to the slide-surface, the computational grid may have steps $O(0.1 \text{ m})$ which is too fine for displayed output. Suppose we divide each prism into stacked layers, say N stacked layers, where the thickness of each layer is much greater than a step in the computational grid. Associated with each layer in a prism are the two outputs, ρ and u. Each prism then has $2N$ outputs plus an H, or $2N+1$ outputs per prism, or $100(2N+1)$ outputs for all 100 prisms from front to tail.

Hold on. That's for only one front position. Suppose we allow the user to retrieve output for 100 front positions from start to stop. Then we have an output matrix $100 \times 100(2N+1)$ for storage, retrieval and graphics. If we chose $N = 10$ we are still well within the capability of storing our output on popular spreadsheets, for example, Q3D uses Excel worksheets and graphics.

But how much output and graphics are within the capability of the user? In other words, how small can we make N? What is the minimum number of layers we need to follow in 100 prisms of swept volume behind 100 front positions? Well, there is strong shear just above the slide surface. Most of us believe and measurements confirm that this leads to separation of flow into faster core layers over slower shear layers. The core layers transfer momentum and energy needed to disaggregate and accelerate entrained and eroded snow.

For wet snow, perhaps we can group the flow into two layers, shear and core, and set $N = 2$. However, dry snow produces spectacular powder clouds which are in layers stacked above the core. Powder clouds reach up to the avalanche-air interface. It appears that as a minimum we need to divide our prisms into three layers, set $N = 3$. That's what we do in Q3D. We divide the swept volume into 100 prisms$\times$3 layers each for a total of 300 prism layers for each chosen front position.

> *9.* Now surely you don't believe a model can predict the amount of snow that will be entrained into the avalanche volume, do you? What user inputs would you suggest to control the amount of entrained snow?

Discussion. The day may come when a 3D model simulates avalanche internals, but a 3D dynamic model can't do everything. In particular, it can't be expected to compute snow stratigraphy in the avalanche path. That's a separate long-term task involving climatology, snow metamorphism, field obs, among other things.

Users of 3D dynamic models are expected to input the mass of all snow which enters the avalanche volume, over the entire swept path from start-zone to L. Some 3D models may distinguish between start-zone snow and snow entrained below the start zone. Q3D takes liberty to call *all* snow which enters the avalanche volume *entrained*, including start-zone snow.

Snow stratigraphy is a myriad of layers and their individual properties. An attempt to input entrainment variables may be little more than a guess. Nonetheless the user can guess, run the model, and be prepared to guess again. Q3D asks the user to supply three entrainment inputs: snow density (kg/m^3), thickness of entrained layer, and a snow wetness index (0.0 to 1.0). Q3D allows the user to vary these three inputs for up to 100 elevations in the Excel input worksheet. Statistically, entrained snow thickness is expected to increase with elevation. Snow wetness and density are expected to decrease with elevation.

It could turn out to meet objective one (simulation of internals) good estimates of entrainmnt are not crucial. Rough guesses may suffice. However, to meet objective two, the better the entrainment estimate, the better the model will serve practical needs. It's expected that inputting extreme values for entrainment thickness would simulate more extreme events.

> *10(a).* Is it essential that 3D models incorporate entrainment?

Discussion. Yes. It is essential despite many complications introduced by entrainment. The user must estimate all entrainable mass. The mass and potential energy stored in entrained snow distributed below the start-zone can't be ignored. It may be twice or thrice the mass of the start-zone snow.

Here is a story of what happened when a primitive model ignored, then included entrainment below the start-zone. Perla, Lied and Kristensen (PLK), attempted a simulation of the Ryggfonn path, Norway. PLK experimented numerically with a 1D particle model which started 500 particles from about a 100 meter start-zone segment. PLK computed particle distribution along CL which had about a 2000 meter horizontal length, and a dam in the runout zone. At first, PLK used two traditional resistance terms: friction resistance μ-term, and dynamic drag u^2-term. There was very little dispersion of the 500 start particles. Next, PLK added a Monte-Carlo term $\pm Ru$ to force dispersion. The result was a bell-shaped distribution, hardly the way avalanches disperse in Nature. Finally, PLK allowed one new particle to be *entrained* every meter the front advanced. Remarkably, distributions changed from bell-shaped to mimic the head-body-tail distributions observed in Nature. That was a gee-whiz moment that made us staunch believers that entrainment from below the start-zone is a necessity, not an option.

Most strongly, no entrainment from below the startzone, then no 3D dynamic model.

> *10(b).* Is it essential that 3D models output the distribution of deposition?

Discussion. Yes it is. Q3D computes, stores and displays deposited mass as well as moving mass. The simulation must confirm conservation of mass. The depostion process does create problems for a continuum model. We'll get to that soon. For now, it's important to understand that entrainment and deposition are not symmetrical processes. The horse eats the grass, the clods left over unused by the horse's digestion don't look like grass.

> *11.* Variables are updated and stored as the front advances. What option would you offer the user? Retreive/Display output for any meter along the path? At 10 meter intervals? At 100 meter intervals?

Discussion. For sure, when the avalanche simulation stops ($u = 0$ in all prisms) we want to display the distribution of deposition from front to tail. For the simulation of avalanche motion, as we mentioned earlier, it's feasible to display stored output for up to 100 front positions (front in motion.) For L $O(10^3\text{m})$, the average interval between the 100 front position outputs is then $O(10$ m). But we may not need or want to review outputs for 100 front positions. Also, we want more flexibility than constant intervals.

> *12.* Perhaps to deal with question 11, first we should ask: "How do we input terrain into our model?" How do we deal with path curvature, inflections such as cliffs and benches, and roughness features at smaller scale? Shall we input the path as a collection of segments? What is a reasonable number of segments along the path centre-line? 10? 100? 1000? Does this choice suggest a possible answer to question 11?

Discussion. Each avalanche path has a unique size and shape. Some are relatively smooth, some highly inflected with cliff bands and benches. Some paths twist and turn, some are relatively straight. Lateral shapes are varied, skinny, fat, pear-shaped, hour-glass shaped....

Consultants are quite good at editing maps, adding in terrain details at a variety of scales. Q3D asks the user to divide CL into a large number of linear segments, not exceeding 100. Firstly, segments are chosen to account for main inflection points of the path (cliff bands and benches;) secondly, to account for left or right lateral deflections of CL, thirdly to account for changes in lateral width, and finally, to divide long, relatively smooth sections of a path. Segment lengths are $O(10$ m) to $O(100$ m). In the simplest case, the top segment (#1) is a slab segment (crown line to *Stauchwall*).

Each time the front enters a new segment, Q3D stores H, ρ, u for all 100 prisms and their layers (front to tail.) Q3D also stores (front to tail) total snow mass in motion, and total snow mass deposited. Therefore, at the end of the simulation ($u = 0$ in all prisms) the user can select a segment number to retrieve and display output (front to tail) for the simulation of the advancing front entering that selected segment. Also, Q3D graphs the deposition for every meter of centre-line, from crown to final front position.

Did you ask what happens at the intersection of the adjoining segments? Does u exiting a segment equal u entering the next segment? That's one tough problem for an advocate of segmenting the path. Some 1D models, PLK for example, attempted to wave off the problem with a simple momentum correction dependent upon change in inclination between adjacent segments. 3D simulations are expected to better handle terrain inflections by taking into account internals such as layer separation. For example, suppose an avalanche interacts with a dam. A simplified three layer simulation (shear, core, and powder layers,) may show that the shear and core layers are stopped by the dam, while the powder layer overruns the dam.

> *13(a).* In earlier one-dimensional models, what user parameters were introduced to model the interaction of snow and terrain?

Discussion. Earlier models (1D and 2D) allowed the user to enter two or three calibration parameters. Some models used two resistive parameters, friction μ and u^2-drag, to match maximum speed and stop position on CL. PLK allowed the user to choose values for a third parameter to help control dispersion. Some models allowed the user to choose dynamic viscosity values to simulate velocity profiles.

> *13(b).* In 3D-modeling should we expect the user to input less, the same number, or more calibration parameters?

Discussion. Of course a minimum of adjustable user parameters is desirable. But there is no free lunch. We display so much more output from our 3D-models compared to 1D-models that we can expect to end up with more than two or three adjustable parameters.

In a sense, entrained snow thickness, density and wetness can be considered three additional parameters which determine energy lost to disaggregate and pelletize swept up snow. In some earlier 1D-models, this lost energy was included in the u^2-drag parameter, adjusted by the user.

How do we handle terrain roughness at smaller scale than segment inflections? Some 1D-models simulated roughness with coulomb friction μ. In it's first stage of development, Q3D asks the user to input μ for each segment. The intention is to replace μ with a parameter which better represents mechanics at the slide-surface. Moreover, Q3D introduces two more calibration parameters to simulate two internal interfaces and to force layering, one to separate shear and core layers; the second to separate core and powder layers. Q3D also requires calibration parameters for it's granular assumptions, as explained below

The long term program, as always, is to replace *ad hoc* parameters with numerical simulation of physical processes. The utility of 3D-models would seem to depend on how successful we are in eliminating *ad hoc* parameters, one by one.

> *14.* The avalanche volume may be modeled as a continuum or as a collection of a very large number of smaller snow masses. What are the pros ands cons of the continuum model? What are the pros and cons following a collection of smaller masses?

Discussion. Here comes our most complex discussion. Please hang on if you've got this far. In a continuum model, we assume (by definition of a continuum) that the expanding avalanche volume V is filled with material whose structure consists of an uncountable number of points. Since we are interested in flow heights H and swept path area S, we take $V = HS$. Inside V we are interested in density ρ and three velocity components u, v, w. If we are clever, we can construct for those variables a set of coupled equations which satisfy conservation of mass, energy, linear momentum, and angular momentum. We must also include entrainment mass flux into V and depositional flux leaving V. And we will need a constitutive equation to describe material properties within V.

If we are very clever, we may be able to numerically solve those equations to find values for S, H, u, v, w, ρ. and then average results within our output grid. Along the way, we may find our solutions include flow separation into layers, and vortices in a coordinate system fixed to say the front of expanding S. In general, that's quite an ambitious approach, but it lends itself to many simplifications which could bring into play a large pedagogy of traditional continuum mechanics. If successful, a continuum model should require considerably less computer time compared to the brute force granular alternative which Q3D adopts.

Q3D assumes V is filled with a large number of long cylindrical rods, radius r, which span the entire path width W. The volume of each rod is $\pi r^2 W$. In Q3D simulations, rods move independently through the rectangular cross-section, remain parallel to each other, always perpendicular to CL, with two velocity components (tangential and normal to the slide surface.) Note that if W is constant along the entire path, rod length would be constant and we would have essentially a 2D simulation with circular areas moving in a plane normal to the slide surface. Simulating a finite and variable path width with rods of variable length W makes Q3D something between a 2D and 3D model, but closer to a

2D circular area simulation of internals because $W \gg r$.

In a full 3D granular simulation, V could be partitioned into spheres with three degrees of freedom, rather than rods with two degrees of freedom. As discussed in *7(a)* output prisms could then be divided into columns for display of 3D variables. A 3D spherical model is far more complex, and computationally intensive.

Even after we solve a set of continuum equations we may add some granular assumptions to deal with impulse forces. While, it may be possible to include entrainment flux into a continuous V, it is not clear how or where to include the depositional flux leaving V. In the continuum approach, layer separation and vortices are often described by a statistical theory of turbulence. A granular approach introduces statistics, randomness, and granular impulse as intrinsic elements from the onset. The granular approach makes it easier to handle deposition. In Q3D, for example, rods with zero speed are dropped from V, and deposited at benches and other concave inflections before the avalanche stops.

PLK obtained reasonable distributions with $O(10^3)$ zero dimensional particles, moving on a 1D centre-line. The computation time of a PLK simulation is almost instantaneous on today's desktops. Q3D uses a much larger number of rods to fill V, and much more computation time dependent upon the number of rods. Q3D assumes we need a large enough number of rods to allow us to compute meaningful statistics.

How does Q3D choose the number of rods, their radii and mass? Q3D makes an arbitrary choice to always entrain 10 new rods for every 0.1 meter of frontal advance. Presumably, this will provide enough rods for output statistics, e.g. averages of u and ρ in all 300 prism layers.

Here's a simplified example of a large event: path length $L = 10^3$m, constant path width $W = 10^2$m, constant entrainment thickness $E = 1.0$ m, and constant entrainment density $\rho_E = 200$ kg/m^3. In this example, the volume of entrained snow is $LWE = 10^5$m^3, and the total entrained mass is $\rho_E LWE = 2 \times 10^7$kg. There are 10^5 total rods in the simulation, the mass of each rod is fixed at 200 kg.

Q3D assumes each rod volume and density, assigned at position of entrainment, remains constant for the entire simulation. This forces a rod radius r to increase where a rod length compresses as a path width narrows. Conversely, r decreases where a rod stretches as a path widens.

Q3D arbitrarily sets the rod density ρ_R to be twice ρ_E. That's an attempt to imitate compaction of entrained snow into higher density pellets. Thus for the assumed entrainment rate of 10 rods per 0.1 m, $r = \sqrt{E/(200\pi)}$. For $E = 1.0$ m, $r \approx 40$ mm.

In its preliminary version, Q3D simply extends the mechanics of PLK, introducing random walks of rods in *both* slope tangential and normal directions. The *Monte-Carlo* term $\pm Ru$ in PLK was not crucially important compared to entrainment in spreading particle distributions parallel to CL. In Q3D, random walks in the slope normal direction are essential to simulate H.

The top layer in a prism will have lower average density compared to middle and bottom layers *only* because it will contain less rods. This is an unsatisfactory feature of Q3D since the powder layer in Nature consists of diffuse dust particles, not fewer high density clods of snow. A continuum model can treat the powder layer using far better constitutive assumptions, such as simulating the powder layer as a dense gas. Perhaps the optimum simulation would combine the best features of the continuum and granular approaches.

Despite its flaws, Q3D simulates and displays internals for the entire event, from start to stop, using reasonable inputs and outputs. Physical correctness of the computational model is important, but so are inputs to, and outputs from the computational model. That's largely the focus of the 14 questions and their discussion.

3D modeling is a very difficult path to climb irrespective of our approach, continuum, granular, or some combination of continuum and granular. We're all mountaineers at heart. We choose to climb because it can be difficult and challenging.

Q3D INPUT/OUTPUT POSSIBILITIES

It is proposed that an EXCEL workbook be associated with each avalanche event.

The workbook could contain perhaps as many as 10 worksheets (WS):

- WS #1... User inputs.
- WS #2... Output of front speed on path profile.
- WS #3... Output of final deposition pattern.
- WS #4... Output of "heights" of moving avalanche.
- WS #5... Output of speeds of moving avalanche.
- WS #6... Output of densities of moving avalanche.
- WS #7... Output of "pressures" of moving avalanche
- WS #8... Storage of outputs used to compute WS #4 thru #7.
- WS #9... Storage of outputs used to compute WS #2 thru #7.
- WS #10... Storage of outputs used to compute WS #2 thru #7.

The opposing page gives an example of WS#1 for a fictitious parabolic path (parab.xls). Similar to the earlier PLK input, the user identifies the profile centerline as a set of segments, determined by horizontal and elevation coordinates. The maximum capability of the model is 100 segments. The user will also have to furnish some estimate of path widths, centerline deflection (left or right), entrainable snow (thickness and density), and path roughness. In the first Q3D version, path roughness will be the familiar parameter μ. Wetness will not be used in the first version.

In addition to the inputs of WS#1, the user will be asked to supply up to four parameters ("USER INPUT PARAMETERS") as the model runs. At least in the early stages of model development, it is envisioned that these parameters will be similar to the calibration inputs used in earlier models such as PLK. The user may change any of the inputs, in which case the workbook will be refreshed.

Q3D solves the dynamics on a grid of approximately 0.1 meter resolution. However, a much coarser grid must be used to display the output. In Q3D, the displayed output is limited to 100 "prisms" as shown on the sketch of the opposing page. These prisms cover the path from the advancing front back to the "tail" which in Q3D is always assumed to be the very top of the path (the top of the slab.) Thus, as the front advances, prism lengths must increase to cover the path from front to tail. The first prism (at the front) has a length of one-meter, the length of prisms 2 through 100 increase logarithmically. In this way, the finest resolution is closest to the front (the head of the avalanche) which is the zone of most interest. Each prism has a length, width and height. The lengths and widths are computed in advance from the path geometry as specified by the user. The prism heights, which can be thought of as local "flow" heights are computed using mass, momentum, force, and energy assumptions. Thus, the model outputs 100 prism heights for each front position. The output is further limited to only those front positions which correspond to either the stop position or the intermediate segment points on WS#1.

Each prism is further divided into three layers as shown on the opposing page:

- Top-layer
- Mid-layer
- Bottom-layer

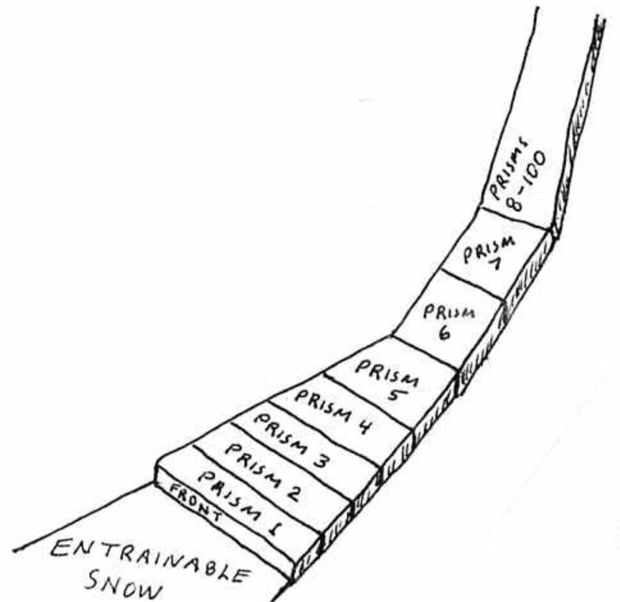
Ideally, these layers would represent the respective: buoyancy layer, core layer, and shear layer. However, the first version of Q3D is too crude for this representation.

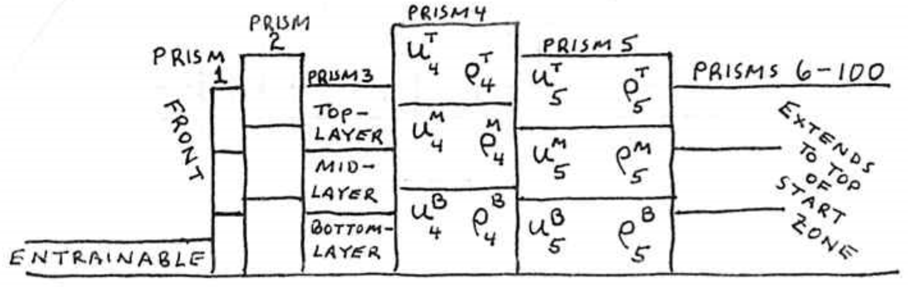
In the first version of Q3D, each layer will have the same thickness, which is simply 1/3 of the prism height. A slope tangential speed u and a density ρ is computed for each layer. Thus, in addition to prism dimensions, the model outputs (for each segment point selected by the user) 600 data points of speed and density (2-variables $\times$ 3-layers $\times$ 100-prisms). For example, on the layer sketch shown on the opposing page, u_4^B means the tangential speed of the bottom-layer in the fourth prism.

Moreover, Q3D must also keep track of the total mass budgets of snow in motion and deposited.

Q3D will compute the dynamics of the event from start to stop, and provide two overviews: the speed profile (WS#2) and deposition distribution (WS#3). If satisfied with these overviews, the user can study the internal structure of the avalanche at intermediate path positions by selecting the segment points of interest. Thus, the next worksheets, WS#4 thru WS#8, are based on the segment point selected by the user. WS#4 thru #8 display the prism and layer variables, front to tail, for the avalanche front entering the selected segment. WS#4 thru WS#8 are refreshed for each segment selected by the user. An example of WS#8 is shown at the bottom of the opposing page. This example is truncated at prism number 10 of the total 100 prisms. WS#8 is a spreadsheet which summarizes the output used to create graphics in WS#4 thru WS#7.

	A	B	C	D	E	F	G	H	I
1	WS #1		**parab.xls**		20-Feb-03		USER INPUTS		
2									
3	SEGMT	PROFILE VIEW		PLAN VIEW		ENTRAINED SNOW			TERRAIN
4	POINTS	HORIZ.	ELEV.	DEFLECTION	WIDTH	THICKNESS	DENSITY	WETNESS	ROUGHNESS
5	(100 max)	(meters)	(meters)	(degrees)	(meters)	(meters)	(kg/m^3)	(0.00 - 1.00)	(0.00 - 1.00)
6				R(+), L(-)					
7									
8	1	0.00	1000.00	0.00	150.00	2.00	200.00	0.00	0.20
9	2	100.00	902.50	0.00	150.00	1.50	200.00	0.00	0.20
10	3	200.00	810.00	-5.00	150.00	1.00	200.00	0.00	0.20
11	4	300.00	722.50	-10.00	100.00	0.75	200.00	0.00	0.20
12	5	400.00	640.00	-15.00	100.00	0.50	200.00	0.00	0.20
13	6	500.00	562.50	-10.00	100.00	0.50	200.00	0.00	0.20
14	7	600.00	490.00	-5.00	75.00	0.50	200.00	0.00	0.20
15	8	700.00	422.50	0.00	75.00	0.50	200.00	0.00	0.20
16	9	800.00	360.00	5.00	75.00	0.50	200.00	0.00	0.20
17	10	900.00	302.50	10.00	50.00	0.50	200.00	0.00	0.20
18	11	1000.00	250.00	15.00	50.00	0.50	200.00	0.00	0.20
19	12	1100.00	202.50	20.00	50.00	0.50	200.00	0.00	0.20
20	13	1200.00	160.00	15.00	50.00	0.50	200.00	0.00	0.20
21	14	1300.00	122.50	10.00	50.00	0.50	200.00	0.00	0.20
22	15	1400.00	90.00	5.00	50.00	0.50	200.00	0.00	0.20
23	16	1500.00	62.50	0.00	75.00	0.50	200.00	0.00	0.20
24	17	1600.00	40.00	-5.00	100.00	0.50	200.00	0.00	0.20
25	18	1700.00	22.50	-10.00	125.00	0.25	200.00	0.00	0.20
26	19	1800.00	10.00	-15.00	150.00	0.25	200.00	0.00	0.20
27	20	1900.00	2.50	-20.00	175.00	0.00	200.00	0.00	0.20
28	21	2000.00	0.00	-20.00	200.00	0.00	200.00	0.00	0.20
29	22	2100.00	2.50	-20.00	225.00	0.00	200.00	0.00	0.20
30	23	2200.00	10.00	-20.00	225.00	0.00	200.00	0.00	0.20
31	24	2300.00	22.50	-20.00	225.00	0.00	200.00	0.00	0.20
32	25	2400.00	40.00	-20.00	225.00	0.00	200.00	0.00	0.20
33	26	2500.00	62.50	-20.00	225.00	0.00	200.00	0.00	0.20
34	27	99999.00							





	A	B	C	D	E
1	WS #8		parab.xls		15 Aug 2003
2					
3					
4					
5	FRONT vs STOP POSITION				
6			FRONT	STOP	
7	PATH DISTANCE (m)		xxxxx	xxxxx	
8	HORIZONTAL (m)		xxxxx	xxxxx	
9	ELEVATION (m)		xxxxx	xxxxx	
10	ALPHA (degrees)		xx.x	xx.x	
11					
12	USER INPUT PARAMETERS				
13	param1 (?mu):			xxxx.xx	
14	param2 (?log m/d):			xxxx.xx	
15	param3 (?):			xxxx.xx	
16	param4 (?):			xxxx.xx	
17					
18	Front arrival time (s):			xxxx.x	
19	Front speed (m/s):			xxx.x	
20	Max. front speed on path (m/s):			xxx.x	
21	Moving mass (kg):			x.xxe+x	
22	Deposited mass (kg):			x.xxe+x	
23	Total deposited at stop (kg):			x.xxe+x	

	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		FRONT ENTERS SEGMENT NUMBER:				xxx								
2														
3														
4				PRISM NUMBER (1 -100). Number 1 prism is at front. Number 100 prism is at top of path.										
5				1	2	3	4	5	6	7	8	9	10	11
6		PATH POSITION (m)		xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	xxxxx	..........>
7		SLOPE ANGLE (degrees)		xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	..........>
8														
9		PRISM LENGTH (m)		*fixed at 1m*	x	x	x	x	x	x	x	x	x	..........>
10		PRISM HEIGHT (m)		xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	xx.x	..........>
11		PRISM WIDTH (m)		xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	..........>
12														
13		MOVING MASS (kg)		x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	..........>
14		DEPOSITION (kg)		x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	x.xxe+x	..........>
15														
16	TOP-	SPEED (m/s)		xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	..........>
17	LAYER	DENSITY (kg / m^3)		xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	..........>
18														
19	MID-	SPEED (m/s)		xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	..........>
20	LAYER	DENSITY (kg / m^3)		xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	..........>
21														
22	BOTTOM-	SPEED (m/s)		xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	xxx.x	..........>
23	LAYER	DENSITY (kg / m^3)		xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	xxx	..........>

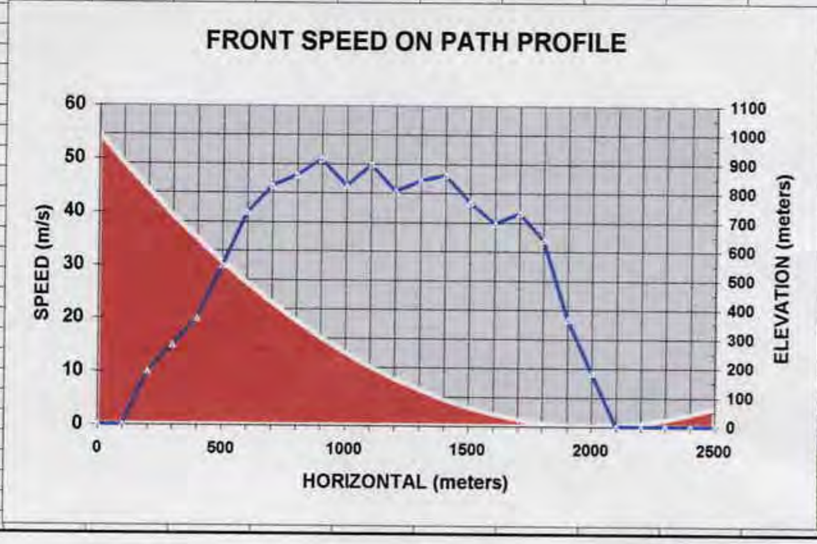
WS#2 provides an overview of the entire motion from start to stop. The user can study this overview and decide if "USER INPUT PARAMETERS" produced a reasonable simulation of maximum speeds and stop position. If not, then the user may rerun the model with new parameters. The workbook will be refreshed.

WS#3 shows the deposition distribution after the avalanche has stopped. Note that the path distances in WS#3, as well as in WS#4 thru #7, are plotted logarithmically in order to maintain highest resolution in the frontal zone of the avalanche. Path distance is measured along the centerline of the avalanche. At some point along this centerline there is a maximum deposition , measured in kilograms per meter of path length. The depositions are plotted relative to this maximum. The user may wish to rerun the model to improve the deposition distribution. The workbook will be refreshed.

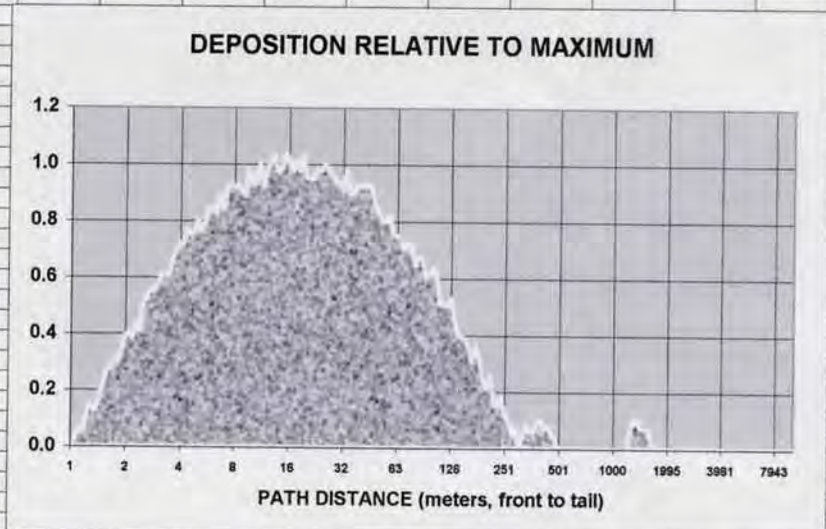
For user convenience, overall model parameters are repeatedly listed to the left of the graphics on all worksheets.

Once satisfied with the overviews as shown on WS#2 and #3, the user may study the motion at any segment point between the start and stop points. WS#4 gives the prism heights of the moving avalanche as it enters the segment selected by the user. Any prism height can only be a rough index of the avalanche "flow height" at that path position. With long term Q3D experience, involving continuing calibration and improved physics, supported by field observation, it is hoped that prism height will be a better index of flow height. In some future Q3D, it may be possible to index separately the thickness of the three layers (bottom–, mid–, and top–layer.) The present assumption that the thickness of each layer is 1/3 the prism height is clearly inadequate.

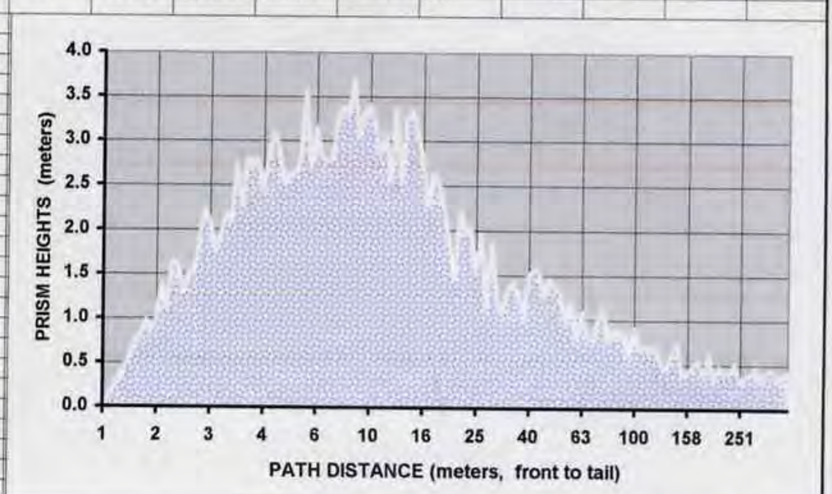
	A	B	C	D	E	F
1	WS # 2.		parab.xls		15 Aug 2003	
2						
3	USER INPUT PARAMETERS					
4	param 1 (? mu):			XXXX.XX		
5	param 2 (? Log m/d):			XXXX.XX		
6	param 3 (?):			XXXX.XX		
7	param 4 (?):			XXXX.XX		
8						
9	STOP POSITION					
10	Path distance (m):			XXXXX		
11	Horizontal (m):			XXXXX		
12	Elevation (m):			XXXXX		
13	Alpha (degrees):			XX.X		
14						
15	Front arrival time to stop (s):			XXXX.X		
16	Total deposited mass (kg):			X.XXe+XX		
17						
18	Max. front speed on path (m/s):			XXX.XX		



	A	B	C	D	E	F
1	WS #3		parab.xls		15 Aug 2003	
2						
3	USER INPUT PARAMETERS					
4	param 1 (? mu):			XXXX.XX		
5	param 2 (? Log m/d):			XXXX.XX		
6	param 3 (?):			XXXX.XX		
7	param 4 (?):			XXXX.XX		
8						
9	STOP POSITION					
10	Path distance (m):			XXXXX		
11	Horizontal (m):			XXXXX		
12	Elevation (m):			XXXXX		
13	Alpha (degrees):			XX.X		
14						
15	Front arrival time to stop (s):			XXXX.X		
16	Total deposited mass (kg):			X.XXe+XX		
17						
18	Max. front speed on path (m/s):			XXX.XX		
19						
20	Maximum deposition (kg/m):			X.XXe+XX		



	A	B	C	D	E	F	G–J	K
1	WS #4		parab.xls		15 Aug 2003		FRONT ENTERS SEGMENT NUMBER:	XXX
5	FRONT vs STOP POSITION							
6			FRONT	STOP				
7	PATH DISTANCE (m)		XXXXX	XXXXX				
8	HORIZONTAL (m)		XXXXX	XXXXX				
9	ELEVATION (m)		XXXXX	XXXXX				
10	ALPHA (degrees)		XX.X	XX.X				
11								
12	USER INPUT PARAMETERS							
13	param1 (?mu):			XXXX.XX				
14	param2 (?log m/d):			XXXX.XX				
15	param3 (?):			XXXX.XX				
16	param4 (?):			XXXX.XX				
17								
18	Front arrival time (s):			XXXX.X				
19	Front speed (m/s):			XXX.X				
20	Max. front speed on path (m/s):			XXX.X				
21	Moving mass (kg):			X.XXe+X				
22	Deposited mass (kg):			X.XXe+X				
23	Total deposited at stop (kg):			X.XXe+X				



WS#5 shows the layer speeds as a function of logarithmic distance from front to tail. It is expected that highest speeds will be found in the mid-layer. The speed trace of the mid-layer could intersect the traces of the bottom and top layers at various slope positions. These intersections may represent rotors which transmit momentum between layers.

WS#6 shows the layer densities as a function of logarithmic distance from front to tail. Density traces could be more stable than the speed traces since density ought to decrease strongly with height above the shear sliding surface. The present version of Q3D (layer thickness = 1/3 × prism height) does not provide a satisfactory simulation of density discontinuity .

WS#7 shows a pressure index for the three layers as a function of logarithmic distance from front to tail. This index is computed simply as ρu^2, using u–values graphed on WS#5 and ρ–values graphed on WS#6. This is probably the simplest pressure index of the many possible indices $f(u, \rho)$. It is easily changed. Q3D could allow the user to input another pressure index from a menu of functions.

WS#4 thru #8 use "PATH DISTANCE" as the independent variable. The alternative is to use arrival time of the prisms at the selected segment point as the independent variable, with the front prism arriving at time $t = 0$. Time graphics could be usefully compared with field measurements from instrumentation installed at fixed path locations. Q3D could allow the user to toggle between PATH DISTANCE and ARRIVAL TIME, with WS#4 thru #8 refreshed by the toggle.

WS#9 and #10 store all outputs for all segments in massive rows and columns without headings. It is expected the user will not be interested in these last two worksheets. They function only to store Q3D outputs which are organized and graphed on WS#2 thru #8.

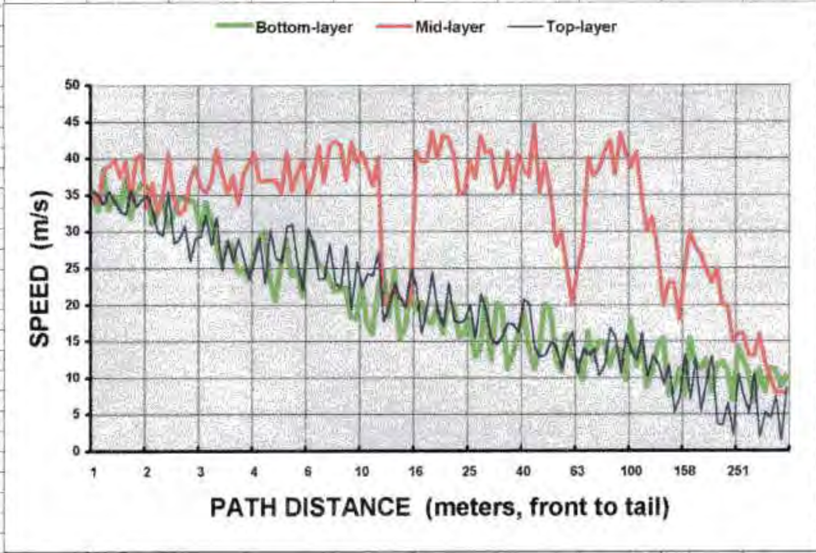
WS #5 parab.xls 15 Aug 2003 FRONT ENTERS SEGMENT NUMBER: xxx

FRONT vs STOP POSITION

	FRONT	STOP
PATH DISTANCE (m)	xxxxx	xxxxx
HORIZONTAL (m)	xxxxx	xxxxx
ELEVATION (m)	xxxxx	xxxxx
ALPHA (degrees)	xx.x	xx.x

USER INPUT PARAMETERS

param1 (?mu):	xxxx.xx
param2 (?log m/d):	xxxx.xx
param3 (?):	xxxx.xx
param4 (?):	xxxx.xx
Front arrival time (s):	xxxx.x
Front speed (m/s):	xxx.x
Max. front speed on path (m/s):	xxx.x
Moving mass (kg):	x.xxe+x
Deposited mass (kg):	x.xxe+x
Total deposited at stop (kg):	x.xxe+x



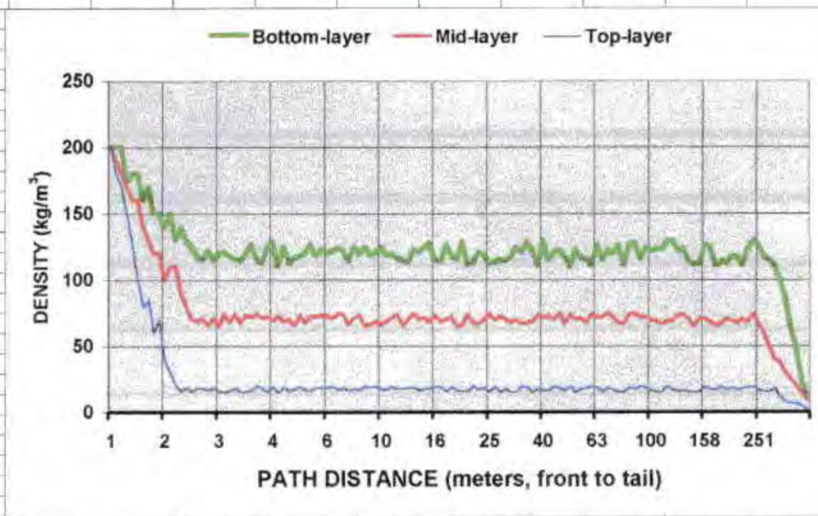
WS #6 parab.xls 15 Aug 2003 FRONT ENTERS SEGMENT NUMBER: xxx

FRONT vs STOP POSITION

	FRONT	STOP
PATH DISTANCE (m)	xxxxx	xxxxx
HORIZONTAL (m)	xxxxx	xxxxx
ELEVATION (m)	xxxxx	xxxxx
ALPHA (degrees)	xx.x	xx.x

USER INPUT PARAMETERS

param1 (?mu):	xxxx.xx
param2 (?log m/d):	xxxx.xx
param3 (?):	xxxx.xx
param4 (?):	xxxx.xx
Front arrival time (s):	xxxx.x
Front speed (m/s):	xxx.x
Max. front speed on path (m/s):	xxx.x
Moving mass (kg):	x.xxe+x
Deposited mass (kg):	x.xxe+x
Total deposited at stop (kg):	x.xxe+x



WS #7 parab.xls 15 Aug 2003 FRONT ENTERS SEGMENT NUMBER: xxx

FRONT vs STOP POSITION

	FRONT	STOP
PATH DISTANCE (m)	xxxxx	xxxxx
HORIZONTAL (m)	xxxxx	xxxxx
ELEVATION (m)	xxxxx	xxxxx
ALPHA (degrees)	xx.x	xx.x

USER INPUT PARAMETERS

param1 (?mu):	xxxx.xx
param2 (?log m/d):	xxxx.xx
param3 (?):	xxxx.xx
param4 (?):	xxxx.xx
Front arrival time (s):	xxxx.x
Front speed (m/s):	xxx.x
Max. front speed on path (m/s):	xxx.x
Moving mass (kg):	x.xxe+x
Deposited mass (kg):	x.xxe+x
Total deposited at stop (kg):	x.xxe+x
Pressure index (kN/m^2):	ρV^2