GSoC 2015 Proposal: Integrating Varying Permeability Model into Subsurface

When did you first hear about Subsurface?

I heard about GSoC from my seniors in the college. I filtered the organisations that gave projects in C and C++. I first read about Subsurface in the GSoC organisations list.

Do you have any diving experience?

No, I do not have an diving experience. However, I would like to learn diving.

What attracted you to Subsurface?

I selected Subsurface because it had interesting projects that required knowledge of C/C++, math and physics. I like both math and physics.

What other open-source tools do you use?

Mozilla Firefox, VLC player, Audacity and GIMP are some open source tools I have used. However, this is the first time I am contributing to an open source project.

Describe any participation by you in the Subsurface community.

I have submitted a couple of patches to the Subsurface code.
A patch to correct the data of diving cylinders stored in the code and a patch to add a safety stop button. During this, I got to interact with many people from the Subsurface community.

Describe any contributions you have made to Subsurface development.

I have submitted patches mentioned above. Creating patches has helped me in getting familiar with the coding style and some parts of the Subsurface code.

Why should Subsurface choose me?

I know sufficient amount of coding, math and physics to be able to complete the project and I am interested in open source coding early on so I would probably be a long term contributor to Subsurface and other open source projects.

Personal Details

- Name: Harshal Jalan
- University: Indian Institute of Technology, Bombay
- E mail: harshal.jalan@gmail.com
- Mobile number: 0091 90041 82007
- IRC nick: harshal
- Google Summer of Code username: harshal_jalan
- Country: India

I am a first year undergraduate pursuing B. Tech in Metallurgical Engineering and Materials Science at Indian Institute of Technology, Bombay. My semester will complete around the end of April and the next semester would start in late July. I have a seven day vacation planned in May but I would continue coding in that period.

I have been coding in embedded C over the past six months as part of my University's student satellite project, Pratham. I have learnt C++ as part of my curriculum in IIT. Though I haven't done open source coding, I am comfortable with collaborative coding through Pratham.

List any other GSoC projects you are applying to.

I would be applying for only one project in GSoC 2015.

Proposal

Name: Implement and integrate VPM algorithm to Subsurface code

Brief explanation: The algorithm currently used in Subsurface is the Buhlmann algorithm ZH-L16. The varying permeability model is an improvement over the Buhlmann model. The task would be to implement the code for the VPM and VPM-B models and integrate them into the subsurface code.

Description

Decompression models are mathematical models that determine if it is safe to ascend from a dive at a given rate.

Buhlmann model ZH-L16 is currently used in the Subsurface code. The algorithm assumes the body to be made of 16 compartments having differing rates of gas absorption. It only considers dissolved gases.

The VPM algorithm adds gas bubbles into the picture by adding a bubble into each compartment. The size of the bubble changes with ascent and descent as well as with passage of time. The goal of the VPM algorithm is to keep the size of the bubble lower than a size where it can cause symptoms of the decompression sickness.

VPM-B is a modification of VPM to add compatibility with Boyle's law. During the planning of complex dives, involving multiple stops while descending as well as use of multiple gases, the VPM model shows significant deviations from the other models. This is corrected by using the Boyle's law to increase bubble radius during ascent.

VPM-B/E adds a layer of conservatism in extreme dives. However, the algorithm is not public and it would be quite difficult to replicate.

Timeline
Till April 27
Continue solving bugs.
Understand the workings of Buhlmann and VPM models more thoroughly.

April 28 to May 5
Understand the way in which the Buhlmann algorithm is currently implemented in the source code and identify which parts can be used for implementing the VPM algorithm. Discuss with mentor and finalise procedure for implementing

the algorithm.
May 6 to May 12 (Week 1)
Setup the basic framework like creating classes for compartments, defining bubble properties like skin and permeability gradient.

May 13 to May 19 (Week 2)
Write or reuse the Buhlmann algorithm from the current source code. The current code may not be directly usable due if corrections have been made in the output to match data.. If so, edit it to match the original Buhlmann algorithm.

May 20 to May 26 (Week 3)
Implement the bubbles in the compartments.
At the end of this week, the code should give an output of the bubble's parameters ie. size and pressure.

May 27 to June 2 (Week 4)
Implement the no-bubble-growth equation. This would give an output showing decompression stops but it would be very conservative.

June 3 to June 16 (Weeks 5 and 6)
Implement a recursive equation that uses the no-bubble-growth output and allows bubbles to expand, keeping them lower than the critical volume.
At this stage, the implementation of VPM would be over. Some work of week 4 may spill over to week 5 but the implementation of VPM should be over by the end of week 6.

June 17 to June 23 (Week 7)
Implement VPM-B by taking Boyle's law into consideration and using it to calculate the increase in bubble size during ascent.

June 24 to June 30 (Week 8)
Testing of the written code against available data. This will also act as a buffer week.

July 1 to July 7 (Week 9)
Start integration into the subsurface code. In the first part, the VPM-B code will replace the Buhlmann algorithm as the code used to calculate decompression stops.

July 8 to July 14 (Week 10)
Add Buhlmann algorithm to the mix, let user the choose the algorithm by

adding a simple radio button interface.

July 15 to July 21 (Week 11)
Debug code using inputs of users and other members in the subsurface community. Polish the code. Do any GSoC paperwork and formalities required.
July 22 to August 17 (Weeks 12, 13, 14 and 15)
Continue debugging. My semester would have begun in this period so this period would act as a buffer. If the integration has completed, try to implement an algorithm similar to VPM-BE using help from mentor and others in the community. If not, the spillover will be completed in this period.

Further work related to the project that I would like to do after GSoC:

- Extension to an algorithm similar to the VPM-B/E algorithm.
- Implementation of the reduced gradient bubble model.