The Journal of Open Source Software

skijumpdesign: A Ski Jump Design Tool for Specified Equivalent Fall Height

Jason K. Moore[1] **and Mont Hubbard**[1]

1 University of California, Davis

DOI: 10.21105/joss.00742

Software

- Review
- Repository
- Archive

Submitted: 18 May 2018
Published: 23 May 2018

Summary

Over the past three decades an evolution has occurred toward freestyle skiing and snowboarding involving aerials in terrain parks at ski resorts hosting dedicated jumping features. Today more than 95% of US ski resorts have such jumps but these rarely, if ever, involve formal or detailed design or engineering. Although usually tested and modified before being opened to the public, they are often simply fabricated based on the past experience of the builder in jump construction. Together with the increase in these jumps has come a concomitant increase in injuries and their very high social costs. Although omitted here, the voluminous epidemiology and financial effects of these injuries are covered in detail in references (Hubbard 2009, McNeil, Hubbard, and Swedberg (2012), Levy et al. (2015), Petrone et al. (2017)).

The likelihood and severity of injury on landing are proportional to the energy dissipated on impact, when the component of velocity of the jumper perpendicular to the snow surface is brought to zero. This energy is naturally measured by the "equivalent fall height" (EFH), defined as the kinetic energy associated with the landing velocity component perpendicular to the landing surface divided by (mg), where (m) is the jumper mass and (g) is the acceleration of gravity.

Past research (Hubbard 2009, A. D. Swedberg (2010), McNeil, Hubbard, and Swedberg (2012), Levy et al. (2015)) has developed a theoretical approach for jump design. It is based on shaping the landing surface so the perpendicular component of landing velocity (and thus impact landing energy and EFH) is controlled to be relatively small everywhere impact is possible. More recent research (Petrone et al. 2017) has presented compelling experimental evidence that these designed jump surfaces embodying low values of EFH are practical to build and, once built, perform as predicted in limiting landing impact. This experimental research has demonstrated that impact on landing can be controlled through design of the shape of the landing surface according to the theory.

Ski resorts have been reluctant, however, to adopt this more engineered approach to jump design, in part due to questions of feasibility, but also because of the somewhat ponderous and complex calculations required. Some recent efforts have been made to develop numerical software to automate these calculations (Levy et al. 2015) that also embodies graphical user interfaces but these have relied on proprietary, closed-source tools and programming environments (MATLAB). The present open source library and online application "skijumpdesign" implemented in Python removes these restrictions and promises to make the design method more widely available to the skiing industry.

The present online application, accessed at http://www.skijjumpdesign.info and seen in Figure 1, allows a relatively unskilled user (e.g. a terrain park manager at a ski resort) to design a ski jump composed of three sections: the approach, landing surface and landing

transition by inputting through sliders four independent design parameters: 1. Parent Slope Angle: The measured downward angle of the parent slope (or a good approximation thereof) where the jump is desired. The designed jump shape is measured from this line. 2. Maximum Approach Length: The maximum distance along the slope above the jump that the jumper can slide to build up speed. The jumper reaches a theoretical maximum speed at the end of this approach and the landing surface shape provides the same impact EFH for all speeds up to and including this maximum achievable (design) speed. 3. Takeoff Angle: The upward angle, relative to horizontal, at the end of the takeoff ramp, a free design parameter. 4. Equivalent Fall Height: The desired equivalent fall height that characterizes landing impact everywhere on this jump.

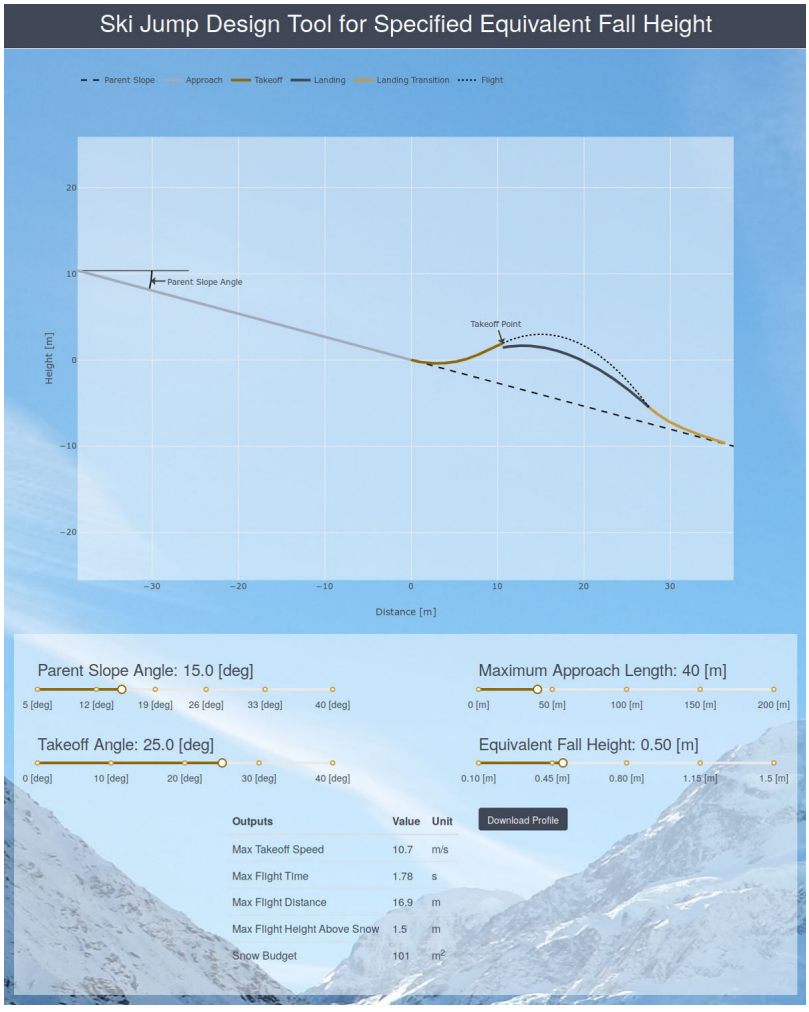

Figure 1: Screenshot of the online application depicting the inputs (4 sliders) and outputs (graph, table, and profile file download).

The output of the program is a graphical display of the total jump surface shape, a table of numerical characteristic results, and a downloadable file describing the jump surface shape in a format useful for jump fabrication. This interface is implemented using Flask (Pallets Project 2018), Dash, and Plotly (Plotly Technologies Inc. 2015) and is suitable for use on any platform that can run a modern web browser. Moreover, we make the interface available online for free use to users that do not want or need to install it on their personal computer.

The online application utilizes a custom server-side Python library for construction of the jump and output parameter calculations. The library's algorithms are implemented using NumPy (Walt, Colbert, and Varoquaux 2011), SciPy (Jones et al. 2001), matplotlib (Hunter 2007), SymPy (Meurer et al. 2017), Cython (Behnel et al. 2011), pycvodes (Dahlgren 2018) and fastcache (Brady 2014). The library provides an application programming interface (API) for simulating planar skiing along arbitrary surface cross sections and two dimensional flight trajectories. This API enables a programmer to use the tools and methods for design needs other than the one posed herein.

Acknowledgements

We acknowledge the assistance of Jim McNeil and Dean Levy who aided in testing the application before release.

References

Behnel, S., R. Bradshaw, C. Citro, L. Dalcin, D.S. Seljebotn, and K. Smith. 2011. "Cython: The Best of Both Worlds." *Computing in Science Engineering* 13 (2):31–39. https://doi.org/10.1109/MCSE.2010.118.

Brady, P. 2014. "fastcache." 2014. https://github.com/pbrady/fastcache.

Dahlgren, Björn. 2018. "Pyodesys: Straightforward Numerical Integration of Ode Systems from Python." *Journal of Open Source Software* 3 (21):490. https://doi.org/10.21105/joss.00490.

Hubbard, Mont. 2009. "Safer Ski Jump Landing Surface Design Limits Normal Velocity at Impact." *Journal of ASTM International* 6 (1):175–83.

Hunter, John D. 2007. "Matplotlib: A 2D Graphics Environment." *Computing in Science & Engineering* 9:90–95. https://doi.org/10.1109/MCSE.2007.55.

Jones, Eric, Travis Oliphant, Pearu Peterson, and others. 2001. "SciPy: Open Source Scientific Tools for Python." http://www.scipy.org/.

Levy, Dean, Mont Hubbard, James A. McNeil, and Andrew Swedberg. 2015. "A Design Rationale for Safer Terrain Park Jumps That Limit Equivalent Fall Height." *Sports Engineering* 18 (4):227–39. https://doi.org/10.1007/s12283-015-0182-6.

McNeil, James A., Mont Hubbard, and Andrew D. Swedberg. 2012. "Designing Tomorrow's Snow Park Jump." *Sports Engineering* 15 (1):1–20. https://doi.org/10.1007/s12283-012-0083-x.

Meurer, Aaron, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, et al. 2017. "SymPy: symbolic Computing in Python." *PeerJ Computer Science* 3 (January):e103. https://doi.org/10.7717/peerj-cs.103.

Pallets Project. 2018. "Flask." 2018. https://github.com/pallets/flask.

Petrone, Nicola, Matteo Cognolato, James A. McNeil, and Mont Hubbard. 2017. "Designing, Building, Measuring, and Testing a Constant Equivalent Fall Height Terrain Park Jump." *Sports Engineering* 20 (4):283–92. https://doi.org/10.1007/s12283-017-0253-y.

Plotly Technologies Inc. 2015. "Plotly: Collaborative Data Science." Montreal, QC: Plotly Technologies Inc. 2015. https://plot.ly.

Swedberg, Andrew Davis. 2010. "Safer Ski Jumps: Design of Landing Surfaces and Clothoidal in-Run Transitions." Master's thesis, Naval Postgraduate School.

Walt, Stéfan van der, S. Chris Colbert, and Gaël Varoquaux. 2011. "The Numpy Array: A Structure for Efficient Numerical Computation." *Computing in Science & Engineering* 13 (2):22–30. https://doi.org/10.1109/MCSE.2011.37.