Computing a Landing Spot on Mars

An Interview with Victor Pankratius

by **Walter Tichy**

Editor's Introduction

The purpose of the Mars rover is in its name—to rove, explore, study Martian geology, look for signs of water, look for signs of life (past or present), etc. However, achieving these and other objectives requires putting the rover down on a suitable landing site, i.e. a site suitable for searching for the desired information and safe to land and function without hindrance or breaking down.

The data for making these decisions comes from prior Mars missions. Selecting a suitable landing site is a complex process typically taking several years. Researchers at MIT's Kavli Institute for Astrophysics and Space Research prototyped a new software that can help NASA mission planners to more rapidly and reliably find landing sites, potentially reducing the total time required to weeks. In this interview, Victor Pankratius, leader of the research team, shares some insight into the project.

Computing a Landing Spot on Mars

An Interview with Victor Pankratius

by Walter Tichy

Walter Tichy (WT): How did you get interested in finding landing sites on Mars? Are you planning a trip?

Victor Pankratius (VP): [Laughing] A trip to Mars would be quite an adventure, but we need to figure out what places would be good for sightseeing...

The idea actually came up accidentally as part of a NASA project I'm leading on computer-aided discovery. We were exploring uses of artificial intelligence in Earth science and realized that we needed new ways of finding geologically interesting sites and events, all while coping with many parameters, constraints, missing data, and uncertainty. There is a strong need in the scientific community to fuse big data from satellite observations from space and sensor networks on Earth, but extracting insights that are physically relevant is difficult. That's why we develop computer-aided discovery techniques that incorporate domain knowledge—in this project that's geophysics knowledge. We explore how domain-aware methods perform in different contexts, including for example volcanoes, earthquakes, and space science.

During an afternoon I wanted to try out a totally different context with my team and see how extensible our tools really were. What if your real-world data does not satisfy common textbooks assumptions? For example, what happens when all you have are sparse, incomplete, or imprecise data sets? After some brainstorming and science fiction sidetracks, the landing site selection came up as an interesting challenge. Site selection had all the problems we were looking for, and NASA's planetary data archives provided data that we could use to test things out. So we started playing with site selection on the Moon first, then went to Mars. At some point we realized that a robust automated approach can have a real impact.

WT: How were landing spots selected before your software did it?

VP: This is a complex process that typically takes several years. It involves various committees including scientists and engineers weighing in on potential missions, science objectives, risks, and constraints. The goal is to find sites that are both scientifically interesting and safe to land. Part of the difficulty is that there are many uncertainties and unknowns, so there is no single best answer. For this reason, candidate sites are discussed in workshops involving a larger community and then narrowed down based on feedback. If you are interested in how the Mars 2020 rover picked the Jezero crater as its landing site, check out <https://www.nasa.gov/press-release/nasa-announces-landing-site-for-mars-2020-rover> and <https://mars.nasa.gov/mars2020/mission/timeline/prelaunch/landing-site-selection>.

WT: What characteristics must a good landing spot have?

VP: A good landing site is easy to get to, easy to land on, and offers opportunities for lots of interesting science. Therefore, we need to break this down to different criteria that we can measure. We can measure Mars' geological features with remote sensing techniques gathered from Mars orbiters or sensors and cameras on Mars rovers. For example, we consider elevation, because a low landing elevation implies a denser atmosphere, which better slows down the descent. But the atmosphere density also depends on uncertain meteorological conditions. Thus, this uncertainty should translate into a range of increasing elevations that become less and less favorable, rather than a strict elevation limit above which the landing becomes impossible. We also consider latitude because the temperature gets colder at higher latitudes, as well as terrain characteristics such as slopes (steeper is worse for a rover to navigate), dust (rovers get stuck there), or geological formations and structures related to oldest known rocks, former deltas, former lakes, volcanoes and others that provide clues to evaluating a site's scientific relevance.

WT: Your software uses fuzzy logic. That's a fairly obscure technique. Can you briefly explain what this is, and how it applies to the terrain on Mars?

VP: Fuzzy logic was proposed by Lotfi A. Zadeh. I met him as a Ph.D. student while attending my first ever academic conference and I was intrigued by his kind personality and his ideas. He suggested that in fuzzy sets, instead of saying an element belongs to a set, yes or no, an element belongs to that set with a certain probability. This is interesting for our Mars application because if you have a number representing a measurement of some sort, this is a way of describing how relevant that number is for your mission context. Fuzzy logic defines all kinds of operators on fuzzy sets, and a whole body of literature has proven their nice properties.

As an example, take a map of Mars as shown on the left in Figure 1. To identify which elevations are good for landing sites, you could simply impose a “hard” threshold. However, with fuzzy logic, those thresholds are more nuanced, allowing you to express that if you are slightly over a threshold it is not that bad, but it gets worse when the difference becomes larger. It is important to have those nuances because sometimes one cannot satisfy all criteria equally well. For each location on the elevation map in Figure 1, the mission planner specifies a value between 0 and 1 indicating how suitable that location would be as a landing site. The middle diagram in Figure 1 shows how that is done: it acts like a filter that maps elevations to the range 0..1. The right diagram shows the resulting range coded in color, where yellow means good, green marginal, and purple unsuitable elevation. The transformation into fuzzy values depends on the particular mission, so our software allows planners to play with different configurations. Figure 2 shows a mapping for latitudes, favoring mid-latitudes.

Figure 1. Fuzzifying elevation criteria: Which elevations are good for landing sites?

Similarly, Figure 2 shows how to look for landing sites within certain latitudes.

Figure 2. Fuzzifying latitude criteria: Which latitudes are good for landing sites?

WT: How are these criteria combined, to obtain a summary value?

VP: To form a “union” of individual fuzzified criteria like the ones above, there are special operators such as the Gamma operator. Figure 3 gives you an intuition about what this operator does. The Gamma operator and others are defined in “Computer-Aided Exploration of the Martian Geology, Earth and Space Science.”

Figure 3. Combining two fuzzy criteria.

WT: Once the Rover has landed, it also needs to go places. Since there are no roads on Mars, you can’t very well apply the classic Dijkstra algorithm for finding short paths. What do you use instead?

VP: You are absolutely right—no roads on Mars, so classical textbook algorithms don’t apply here. Therefore, our approach uses an algorithm known as “Fast Marching” to chart out paths that a rover can take over a given terrain (see Figure 4, right). Fast marching typically calculates the propagation of a front, such as how fast a front of wind reaches a shore if traveling at a given speed. For the first time, we applied fast marching in a totally different context -- to

compute a rover's travel time as it travels from a starting point to a geological structure of interest.

Figure 4. Deriving possible rover paths.

If you pick two locations, this method computes travel times rather than distances. The idea is that if a rover lands close to a target it doesn't mean that it will be able to reach it quickly. There might be obstacles or shadows at certain times that don't allow recharging your batteries. Also, some rovers can drive faster if you go straight and slow down when taking turns. All of these issues are great classroom examples to explain non-Euclidean metrics, by the way.

Intuitively, you might want to think of dropping a stone in a lake as an analogy to understanding how the fast marching method computes a front of travel times. Ripples propagate from the location where you dropped the stone. If there are no obstacles, the ripples are circular. However, a rock in the way would change the shape of the propagating waves. On Mars, high slopes and dust are like these obstacles. We compute the front to figure out the driving time to reach one location from another. We can also analyze the path itself and count the types of geological targets it would hit. This step allows us to create aggregate statistics for many location pairs and paths that are navigable by current rovers, which provides new pieces of information for the science value of the mission.

WT: Is there a way to evaluate the quality of the results? How long does the software run to deliver them? And once it has found candidates, how is the final decision made?

VP: One way to evaluate the quality of results is to compare outputs with past landing sites selected by missions and committees. The paper shows this information on the maps in the figures if you are interested. The philosophy of our approach is to keep the human in the loop, so the software acts more like a decision support system that allows people to play with parameters and see the outcomes in different scenarios. So the final decision is always made by humans. It is hard to give you a general number on runtime, but it's safe to say that computing time is insignificant compared to the number of years humans can spend to coordinate and make a decision. In addition, the software can give humans a broader selection of options for science scenarios they might want to consider.

WT: Will landing site selection be totally automated one day?

VP: This is a great question. Let me start by saying that it depends on the context whether full automation performs better than humans or not. So let me outline potential future directions where increased autonomy is important for creating new capabilities that we have never had before. More autonomy in the landing decision can enable totally new missions, like landing on Europa, for example. The underlying issue is that we want to be able to go to places that are far away where signals might take too long to reach any human, or places that have limited time windows for communication. For some of those places we have barely any maps or actionable information that we could use in planning ahead. Harsh environments require quick adaptation for missions to succeed. So in a way, our ability to expand our exploration capabilities depends on progress in these areas. In any case, however, the criteria and operating parameters will still be defined by humans one way or the other, even within a system with increased autonomy.

WT: *The Washington Post* recently carried an [article about growing grapes on Mars](#). What do you think about that?

VP: I love the entrepreneurial spirit! I hope it is red wine. Is it supposed to keep the astronauts happy?

(Editor's note: According to *The Washington Post*, prospective space vintners think that white grape varieties are preferable, because they are hardier.)

Acknowledgment

Figures 1 through 4 are adapted from Guillaume Rongier and Victor Pankratius' "Computer-aided Exploration of the Martian Geology," with permission from *Earth and Science Journal*.

Further Reading

Jennifer Chu. [Software finds the best way to stick a Mars landing](#). *MIT News* (Sept. 26, 2018).

Guillaume Rongier and Victor Pankratius. Computer-aided exploration of the Martian geology. *Earth and Space Science* 5, 8 (August 2018), 393—407. <https://doi.org/10.1029/2018EA000406>

About the Author

Walter Tichy has been professor of Computer Science at Karlsruhe Institute of Technology (formerly University Karlsruhe), Germany, since 1986. His major interests are software engineering and parallel computing. You can read more about him at www.ipd.uka.de/Tichy.

DOI: 10.1145/3323995