1. **In terms a general audience would understand, describe an important, outstanding problem in computer science, applied mathematics or statistics that you would like to pursue in your research. (1/3)**
2. **Discuss the potential impact of your research on your field. How would it advance high performance computing in general? (1/3)**
3. **Provide an example of how it would advance a science or engineering application area or areas. (1/3)**

Linear algebraic calculations are a near-universal subproblem in most areas of computational science: from the simulation of fluid flow around airplane wings to the study of electronic properties of novel materials to discovering patterns in social networks to detect bad actors. These linear algebraic problems may be as simple as a high school algebra problems--solve these two equations for x and y--scaled up to gargantuan proportions like solving for 10 billion unknowns in 10 billion equations. While the area of computational linear algebra has been extensively studied, new application areas (machine learning and data science), new discretization techniques (novel high-order methods and methods producing non-square or dense matrices), and new advances in hardware (GPUs and exascale computing) require novel algorithms to effectively solve linear algebraic subproblems for our most important scientific calculations. Luckily, exciting research combining randomization, low-rank structure, and matrix-free methods shows promise to meet our need for novel algorithms, and these are the techniques I hope to use and extend in my PhD work. What I find so exciting about computational linear algebra is that algorithmic breakthroughs have the possibility of substantially accelerating high-performance computations in a numerous number of application areas. While I'm excited by several application areas where big advances in linear algebraic algorithms could prove revolutionary, one area that is especially near and dear to my heart is first-principles molecular dynamics (FPMD) simulations. As I learned during my work at Sandia National Labs, FPMD calculations, which quantum mechanically model the electronic structure of materials, are super useful but super slow. Thankfully, the work of many researchers (many at DOE labs) have developed linear-algebraic algorithms which vastly speed up these calculations, but these methods are still much slower than less accurate but faster classical MD. Developing new methods using randomized matrix-free low-rank approximations to rapidly perform these calculations could prove revolutionary, allowing for a best-of-both-worlds fast, quantum-mechanically accurate material modeling.

1. **Discuss the role of high performance computing in your research. (1/2)**
2. **How will you demonstrate the success of your research? (1/2)**

The linear algebraic problems of today are of such massive scale that supercomputers are essential to even storing the entire problem, not even to mention solving it. Problems routinely solved by scientists at DOE labs frequently enter the many billions of equations and unknowns and when algorithmic advances are made to accelerate these computations, domain scientists excitedly use this new performance to study even larger or more intricate models. As such, novel fast algorithms for linear algebra problems are of great interest to nearly the entire HPC community. The development and deployment of new large-scale linear algebra algorithms is a many-year process, where the algorithm is usually first designed and analyzed for serial computation before research is done on parallelizing it. Once parallelized, significant further gains can unusually be achieved by specialized optimizations to fully take advantage of the hardware. I have the energy and aspiration to design and develop a method and see it through all the stages of this process (and I will go as far as I can during the course of my PhD). However, I believe my research will still have been a terrific success if I am able to design a novel method which substantially improves upon existing methods for a particular problem from a specific application (such as electronic structure calculations, data mining, or another) and parallelize it (or at least outline how it will be parallelizable). To me, establishing that a method is an improvement upon existing methods entails either fast runtimes or better runtime scaling behavior than existing methods on actual iterations of the problem (ideally in both serial and parallel), as well as accompanying theoretical work proving correctness, speed, and accuracy of the new method. Once the method is designed and parallelized, I would hope to have time remaining to work on high-performance optimizations and producing well-documented, and publicly available source code of the algorithm.

Describe how the courses listed in your planned program of study would help prepare you to address the challenges you have described in questions 1 and 2. Discuss your rationale for choosing these courses. How will the science or engineering application courses you have selected impact your research?

Essential to doing research in computational linear algebra is a strong grounding in classical mathematics: linear algebra, mathematical analysis, and probability theory (for randomized algorithms, which are becoming an increasingly important part of the field). I already have very strong coursework in these areas from my undergraduate degree, but delving deeper on these topics again at an extra-challenging school like Caltech will help me further develop and hone these core skills. My graduate-level special topics courses on matrix computations, fast matrix algorithms, and sparse matrix algorithms are equally essential, and I look forward to continuing to grow within my specific field by topics courses at the university I attend (such as ACM/IDS 204 and ACM 217 at Caltech). Important to the high-performance implementation of these algorithms is parallel and GPU programming. I believe that Caltech's courses in GPU and distributed computing should give me a foundation in HPC computing, which I will supplement (as I have already done) with independent study of parallel programming using MPI and OpenMP. As linear algebra has such wide-ranging applications, the specific application courses that will best suit me will depend considerably on what school I attend and what line of research I pursue with the direction of my advisor. For this reason, I selected three dramatically different engineering courses for my POS, a machine learning class, an advanced signal processing class, and fluid mechanics class. I believe the combination of these three topics will help give me a well-rounded introduction to both “classical” science and engineering applications of linear algebra (by solving discretized PDEs from a continuum mechanics problem) and emerging applications in data mining and machine learning.