Abstract: As the scale of parallel computing systems increases, resilience becomes increasingly important. Traditional fault-tolerance mechanisms that employ application-initiated checkpoint/restart can be I/O intensive. Accordingly, considering the continuously growing gap between the performance of processors and storage devices in high-end computing (HEC) systems, in order to sustain application throughput, advances in I/O performance are essential. Our research addresses this problem with a three-pronged approach.

1. *Optimization of I/O resource usage*: Development of a methodology for coordinating productive and defensive I/O (associated with checkpoint/restart), while optimizing system performance. This coordination aims to optimize the use of I/O-related resources by controlling the volume of defensive I/O. To do this, we will use mathematical models to guide the selection of application checkpoint frequencies, which effect both execution times and the number of I/O operations generated.
2. *Differentiated I/O*: Design, development, and evaluation of algorithms that provide differentiated RAID I/O service (2011 paper) and parallel file system service, with little impact on aggregate throughput. Differentiated I/O entails proportional service in terms of workload weights in addition to performance isolation, which has a high potential of translating to predictable application performance. Such predictability can facilitate effective I/O coordination and, therefore, improve HEC system throughput.
3. *Analysis of failure data for model parameters:* Analysis of publicly-available failure data, including data from the Computer Failure Data Repository (CFDR), RAS logs from Blue Gene/P Intrepid, and Blue Gene/L event logs. Through the use of statistical tools, data mining, and neural networks, we aim to determine realistic parameters for the mathematical models that will form the basis of our defensive I/O coordination algorithm.

Key objectives of the proposed research:

- Provide differentiated I/O service with little change in aggregate throughput.
- Coordinate productive and defensive I/O.
- Increase meaningful utilization of allocated computing resources to enhance system performance.
- Extend scalability of checkpoint/restart fault management.
- Reduce I/O system stress and resultant failures.

The body of research resulting from this work is expected to answer the following questions for HPC application developers:

a. *How infrequently can checkpoint operations be performed without adversely affecting application execution time?*
b. *Is it possible to scale application performance even when employing traditional fault-tolerance strategies that employ periodic checkpointing?*
c. *Given information about a set of periodic-checkpointing applications that are expected to concurrently execute on an HPC system, how can the applications be modified (including the selection of checkpoint intervals) to minimize contention at the shared I/O system?*