An Automatic Actors to Threads Mapping Technique for JVM-based Actor Frameworks

Ganesha Upadhyaya and Hridesh Rajan
(ganeshau,hridesh)@iastate.edu

Problem: Mapping Actors to JVM threads is important to efficiently utilize CPU cores and it is challenging because
- more actors than JVM threads,
- actors are short-lived,
- OS-Scheduler maps JVM threads to Cores

Solution outline
1) analyze actor characteristics and communication behaviors
2) represent them in Actor Characteristic Vector (cVector)
3) apply mapping function to determine the execution policy

Solution: An Automatic Actors to JVM threads mapping technique can help programmers focus on their design

Insight:
Use Actor Characteristics and Communication behaviors

Approach:
Use static analyses to infer actor characteristics and communication behaviors to assign execution policies

Execution Policy defines how actor’s messages are processed

Input

Output

Key

Results

Large improvements for actor programs with sub-optimal performance benefits

Small or no improvement for data parallel actor programs

This work was supported in part by the NSF under grants CCF-08-46059, CCF-11-17937, and CCF-14-23370.