Comparing Actor System Topologies and Parameters Using *BeCoMe*

Marco GREBE [a,b], Tilman LACKO [a] and Rita LOOGEN [b]
[a] *Volkswagen AG, Wolfsburg, Germany*
[b] *Philipps-Universität Marburg, Germany*

Abstract. Benchmarking and rating constantly running, distributed, message based systems is very difficult. Time based benchmarks require high effort and are very error prone and expensive because they need dedicated hardware. Instead we propose to use the number of messages in such systems as an abstract layer which is independent from physical clusters but nevertheless allows to infer physical behaviour using hardware specific cost vectors. The number of messages and cost vectors can be systematically extracted from a running instance of a message based system. We show how results gained using a traditional benchmark and those inferred using our technique correlate using two real world examples.

Keywords. Geographic Maps, Parallel Computing, Message Passing, Actor Model

1. Introduction

Time based benchmarking of distributed continuously working systems is hard to perform. A lot of preparation work and additional parts, for example termination detection, are needed to compare different parametrizations, while the needed hardware has to be reserved exclusively for the tests. We propose a system called *BeCoMe* for assessing distributed message passing systems by capturing the messages and using their carefully chosen type information to construct a model of the system behaviour, resource consumption, and scaling behaviour for changing parameters. The model can be used to infer the impact of additional nodes or changing input compositions onto the systems performance.

We show the advantages of *BeCoMe* by analysing two different real world systems. Both systems are used to merge observations made by the sensors of cars into a geographic map of objects in the real world. Traditional benchmarks times the duration to process a fixed number of measurements for different numbers of nodes and parameters duration. This generates comparable values, but underlying rules which lead result are a black box. Using direct mathematical formulas to model the amount of work needed to process measurements does show the rules. The modelling is very complex, since the algorithms structure has to be translated into a comparable form. Further it is error prone: formulas are build to model what is expected to happen instead of what is happening. Our approach is located between those two variants.

In pure message passing systems all information between different threads is exchanged using (asynchronous) messages. Especially distribution of work, control of a