One of the most interesting aspects of parallel and distributed architectures is to design the algorithms that best suit the design of the underlying computational architecture. This session considers three papers that cover different application areas.

The first paper, “Increasing the Adaptivity of Routing Algorithms for k-ary n-cubes” by E. Baydal, P. Lopez and J. Duato considers the behaviour of both deadlock avoidance and recovery adaptive routing algorithms exploiting this increased routing flexibility, comparing them with previous proposals, in order to evaluate the contribution of the additional routing freedom on network performance. Simulation results show that this simple improvement in the routing algorithm allows improvements to be achieved in throughput up to 45% in networks with low radix, for the uniform distribution of message destinations.

“Removing the Latency Overhead of the ITB Mechanism in COWs with Source Routing” by J. Flich, M. P. Malumbres, P. Lopez and J. Duato is the second paper in the session and analyses in detail the latency overhead of ITBs, proposing several mechanisms in order to reduce, hide, and remove it. The paper proposes three mechanisms that will try to overcome the latency penalty. All the mechanisms are simple and can be easily implemented. Results show a very good behaviour of the proposed mechanisms, reducing considerably, and even removing the latency overhead.

The third paper, entitled “Solving Non-Smooth Unconstrained Optimization Problem with LAMGAC in a LAN-WLAN Grid Domain” by E. Macías, A. Suárez 1, C. N. Ojeda-Guerra and L. Gómez, considers the performance of WLANs as a resource for Grid Computing. The paper presents a library based on LAM/MPI named LAMGAC for programming in a LAN:WLAN cluster and parallelizing an algorithm that finds the global minimum of a nonlinear real valued continuous function. The algorithm uses a strategy based on the division of the domain into small boxes and locates the extreme by means of a multiple start algorithm (MRS). The local minimizer is carried out by means of the steepest descent and the DFP method. The novelty of this approach is that we can vary the parallel virtual machine in runtime.