
Type of paper: radical solution

Correlation to the real-world setting: Developers are documenting architectures informally. However, these informal architectures can’t be verified for consistency.

What is the hypothesis: Connectors are just as important as the interfaces of the modules that they connect. The architectural validation language defined in this paper can be translated into CSP, which can then be validated to show compatibility by searching for deadlocks.

Validation: Demonstrate the approach on a simple example. Show that the validation can point to non-obvious problems in the example. The paper also includes some persuasion to argue that this approach is better than other solutions.

Result: Local checks can ensure compatibility for any ports implementing the protocols. Running checks on even simple ports indicated problems with the specification of the ports.

Do you believe the result, and why? I believe the point that imports and other module specifications in programming languages do not substitute for this type of model. Connectors also seem to be a good idea. I’m not convinced that this approach will scale to validate certain properties of connectors, such as data ordering and deadlock avoidance.