Aggregate Computing

Mirko Viroli
2019/02/01 16:05
Table of Contents

Aggregate Computing ................................................................................................................................................................. 3
Short Intro ............................................................................................................................................................................... 3
People ...................................................................................................................................................................................... 3
Stack of models and technologies ........................................................................................................................................... 3
News ........................................................................................................................................................................................ 3
Bibliography primer ................................................................................................................................................................. 3
Bibliography categories ........................................................................................................................................................... 3
Aggregate Computing

Short Intro

Aggregate programming provides an alternate approach to the standard device-centered development methodology, aiming to dramatically simplify the design, creation, and maintenance of complex and large scale software systems, specifically in the context of IoT, cyber-physical systems, pervasive computing, robotic swarms, and large-scale situated systems in general. Under this approach, the reference computing machine is no longer the single device, but is instead the aggregate collection of devices that cooperatively carry out a computational process: the details of behaviour, position and number of devices are largely abstracted away, to be replaced with a space-filling computational environment. Hence, with aggregate programming the availability of large, heterogeneous sets of devices becomes less of a concern and more an opportunity to increase the quality (e.g., soundness, stability, efficacy). In fact, standard software engineering (and software quality) best practices can then be applied at the aggregate level, there including: modularity/compositionality of specification/libraries, tractability of property analysis, and so on.

People

Close researchers involved in this research activity include:

- Università di Bologna, Italy: Mirko Viroli, Danilo Pianini, Roberto Casadei
- BBN Raytheon Technologies, USA: Jake Beal
- Università di Torino, Italy: Ferruccio Damiani, Giorgio Audrito

Stack of models and technologies

- Computational fields: maps from devices to values; in our approach "everything is a field"
- Computational Field Calculus: a tiny language, amenable to formal treatment, describing computations over fields
- Protelis language: a Java-oriented, Java-enabled programming language to construct aggregate computations based on the computational field calculus
- Alchemist: a fast simulator, with ability (among the others) of running Protelis, other than importing maps, GPS traces, images and so on
- ScaFi: a Scala-based DSL and library for building simulators and systems adopting the Aggregated Computing view

News

- Paper accepted at ACM Transactions on Computational Logic: TOCL2019
- Paper published at ACM Transactions on Computer Modeling and Simulation: TOMACS2018
- Paper published at ACM Transactions on Adaptive and Autonomous Systems: TAAS2017
- Presentation given locally at the "Open Your Mind" seminars series: OpenYourMind-AggregateComputing.pdf
- Paper accepted at SASO 2015! AggregateComputingSASO2015
- I'll be giving a keynote speech at the "Collective Adaptation in very large scale UBICOMP: The Superorganism of Massive Collective Wearables" UBICOMP workshop, sponsored by FOCAS (http://focas.eu/towards-superorganism-wearables/)

Bibliography primer

- Flagship paper
- Theory
- Properties
- Overview of research
- Universality

Bibliography categories

Background

- Description and composition of bio-inspired design patterns: a complete overview. A catalogue of patterns of self-organisation
- Chemical-oriented Simulation of Computational Systems with Alchemist: Presentation of the architecture of Alchemist simulator
- Organizing the Aggregate: Languages for Spatial Computing: Survey on previous aggregate computing-related attempts
Themes - Aggregate Computing

Field Calculus
- A type-sound calculus of computational fields: Presentation of the computational field calculus (extended version of FieldCalculusFOCLASA2013)
- Type-based Self-stabilisation for Computational Fields: A fragment of field calculus proved self-stabilising (extended version of SpatialCoordination2014)
- Towards a Unified Model of Spatial Computing: A universality result for the computational field calculus
- Code Mobility Meets Self-organisation: A Higher-Order Calculus of Computational Fields: Adding higher-order features to computational field calculus, key to support openness

Aggregate Computing Methodology
- Building Blocks for Aggregate Programming of Self-Organising Applications: Proposing a set of building blocks on top of field calculus, to scale with complexity
- Space-Time Programming: A survey in the potentials and challenges of aggregate computing (more generally, space-time computing)
- Aggregate Programming for the Internet of Things: The flagship publication of Aggregate Computing, containing a rather accessible introduction to the general topic