Static analysis of incremental propagation graphs with process algebra
Théo Le Calvar, Fabien Chhel, Frédéric Jouault, Frédéric Saubion

To cite this version:
Théo Le Calvar, Fabien Chhel, Frédéric Jouault, Frédéric Saubion. Static analysis of incremental propagation graphs with process algebra. ACM/IEEE 21st International Conference on Model Driven Engineering Languages and Systems (MODELS) 2018, Oct 2018, Copenhague, Denmark. 2018. hal-02715065

HAL Id: hal-02715065
https://univ-angers.hal.science/hal-02715065
Submitted on 26 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Static analysis of incremental propagation graphs with process algebra
Théo Le Calvar$^{1,2}$, Fabien Chhel$^{2}$, Frédéric Jouault$^{2}$, Frédéric Saubion$^{1}$

$^1$ LERIA, Université d’Angers, Angers, France. $^2$ ERIS Team, ESEO Group, Angers, France

Abstract

Active Operations are a set of operations that can be composed to build incremental bidirectional OCL-like expressions on collections. The current implementation of active operations (AOF) relies on the Observer design pattern to propagate changes from each operation to its successors. These relations form an implicit directed acyclic propagation graph. In this work we present a new relational notation to describe propagation graphs. Along with this notation, we also present a new static analysis method of the propagation graph based on process algebra. This new method enables optimizations of the propagation graph not achievable with previous approaches, such as detection of parallelizable sections of the propagation graph or cache optimizations in specific situations.

Process Algebra

Process Algebra [2, 4] is a formalism used to formally describe process behaviors. It consists of a small sets of concepts and operators such as:

- Actions: $a, b$ are simple actions, $\delta$ is a special action that corresponds to a failure.
- Sequential: $a; b$, a then $b$.
- Alternative: $a + b$, $a$ or $b$.
- Parallel: $\pi(a, b)$, $a$ and $b$ in parallel.
- Communication: parallel processes $a$ and $b$ can communicate if $a \parallel b$ is defined.
- Encapsulation: $\exists X \in H \sum_{\delta \in \text{Start}} \partial(X) \cap R2ACP(o) = 0$

Example of transformation

Using process algebra to describe AOF expressions offers several interesting analysis:

- The formula can be used to check if an operation evaluation ordering is correct.
- The formula can be used to generate correct orderings (operation evaluation).
- One specialized ordering per input of the expression.
- Parallelizable sections can be inferred from the formula.
- Possible detection of useless operation caches.
- Recursive formulas could be used to model expressions with cycles.
- Process algebra has tools that could be used to optimize propagation graphs.

References