

# AntGen EDSL

J. Alpuim   L.T. van Binsbergen   J.P. Pizani Flor

Department of Information and Computing Sciences, Utrecht University

January 31, 2013

## The Application

- Current strategy
- Improvements

## The Library

- AntStrategies
- AntMoves

## The Language (EDSL)

- Imperative-ish constructs
- Translation to Ant assembly

## Compiling to Assembly

- Composing assembly blocks
- Program transformations



# Table of Contents

## The Application

- Current strategy
- Improvements

## The Library

- AntStrategies
- AntMoves

## The Language (EDSL)

- Imperative-ish constructs
- Translation to Ant assembly

## Compiling to Assembly

- Composing assembly blocks
- Program transformations

## The Application

- Current strategy
- Improvements

## The Library

- AntStrategies
- AntMoves

## The Language (EDSL)

- Imperative-ish constructs
- Translation to Ant assembly

## Compiling to Assembly

- Composing assembly blocks
- Program transformations



# Current strategy

Our main strategy consists of

- ▶ Draw “highways” from the corners of the nest
- ▶ Look for food using a “ricochet” movement
- ▶ Use “highways” to come back to the nest

## The Application

Current strategy

Improvements

## The Library

AntStrategies

AntMoves

## The Language (EDSL)

Imperative-ish  
constructs

Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks

Program  
transformations



# Possible improvements

- ▶ Protecting corners of the nest
- ▶ Drawing local roads
- ▶ Avoiding congestions

## The Application

Current strategy

**Improvements**

## The Library

AntStrategies

AntMoves

## The Language (EDSL)

Imperative-ish  
constructs

Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks

Program  
transformations



# AntStrategies

Some examples of high-level strategies include:

- ▶ Ricochet walk
- ▶ Random walk
- ▶ Follow any pheromone track
- ▶ Draw any pheromone track

## The Application

Current strategy  
Improvements

## The Library

**AntStrategies**  
AntMoves

## The Language (EDSL)

Imperative-ish  
constructs  
Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks  
Program  
transformations



# Ant moves

Some examples of simple moves include:

- ▶ Safe move
- ▶ Random choices
- ▶ Interleaving strategies

## The Application

Current strategy  
Improvements

## The Library

AntStrategies  
**AntMoves**

## The Language (EDSL)

Imperative-ish  
constructs  
Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks  
Program  
transformations



# Imperative-ish constructs

We developed some imperative-like constructs in order to get a well defined flow of control for each program:

- ▶ Sequence
- ▶ Loops (while, forever)
- ▶ Conditionals (ifThen, ifThenElse, side-effecting test, case)
- ▶ Boolean operators

## The Application

Current strategy  
Improvements

## The Library

AntStrategies  
AntMoves

## The Language (EDSL)

### Imperative-ish constructs

Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks  
Program  
transformations



# Translation to Ant assembly

- ▶ EDSL datatypes
- ▶ Semantic functions
- ▶ Boolean algebra

## The Application

Current strategy  
Improvements

## The Library

AntStrategies  
AntMoves

## The Language (EDSL)

Imperative-ish  
constructs

Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks  
Program  
transformations





# Composing assembly blocks

- ▶ One composing function per imperative construct
- ▶ Using supply monad to get ant states
- ▶ Map AntState AntInstruction
  - with an initial and final state

## The Application

Current strategy  
Improvements

## The Library

AntStrategies  
AntMoves

## The Language (EDSL)

Imperative-ish constructs  
Translation to Ant assembly

## Compiling to Assembly

Composing assembly blocks  
Program transformations



# Program transformations

We have used two functions to get a full-working ant assembly code:

- ▶ Ghostbusters
- ▶ Keys to line numbers (ordering the map to  $[0, n - 1]$ )

## The Application

Current strategy  
Improvements

## The Library

AntStrategies  
AntMoves

## The Language (EDSL)

Imperative-ish  
constructs  
Translation to Ant  
assembly

## Compiling to Assembly

Composing  
assembly blocks

**Program  
transformations**

