

# Soft Constraints for Large Self-organizing Systems

## Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Motivation . . . . .	5
1.2	Contributions of the Thesis . . . . .	5
<b>2</b>	<b>Application Scenarios</b>	<b>7</b>
2.1	Distributed Power Systems . . . . .	7
2.1.1	The Unit Commitment Problem . . . . .	7
2.1.2	Complexity of the Problem . . . . .	7
2.1.3	Related Work . . . . .	7
2.2	Self-organizing Robotic Systems . . . . .	7
2.2.1	Task Allocation in Reconfigurable Swarms [ECAS'16] . .	7
2.2.2	Self-organizing Resource-Flow Systems . . . . .	7
2.2.3	Related Work . . . . .	7
2.3	Preference-oriented Adaptive Systems . . . . .	7
2.3.1	Exam Appointment Scheduling . . . . .	7
2.3.2	Mentor Matching [Constraints'17]* . . . . .	7
2.3.3	Multi-User Multi-Display Exhibitions [ModRef'15]* . . .	7
2.3.4	Related Work . . . . .	7
<b>I</b>	<b>Soft Constraints for Corridors of Optimal Behavior</b>	<b>9</b>
<b>3</b>	<b>Foundations</b>	<b>11</b>
3.1	Self-organization – Designing Systems to Adapt . . . . .	12
3.2	Controlling Emergence – The Restore Invariant Approach . . . .	12
3.2.1	Corridors of Correct Behavior . . . . .	12
3.2.2	Classical Constraint Programming for Combinatorial Problems . . . . .	12
3.3	Over-Constrainedness in Practical Problems – From Correctness to Optimality . . . . .	12
3.3.1	Partial Constraint Satisfaction . . . . .	12
3.3.2	Specific Soft Constraint Formalisms . . . . .	12
3.3.3	Algebraic Structures for Soft Constraints . . . . .	12
3.4	Algorithms to Solve Soft Constraint Problems . . . . .	12
3.4.1	Branch-and-Bound Search . . . . .	12

3.4.2	Large-Neighborhood Search . . . . .	12
3.4.3	Optimization in Constraint Programming Solvers . . . . .	12
3.5	Modeling Languages . . . . .	12
3.5.1	MiniZinc and MiniSearch . . . . .	12
3.5.2	Essence . . . . .	12
3.5.3	Numberjack . . . . .	12
<b>4</b>	<b>Soft Constraints with Constraint Preferences</b>	<b>13</b>
4.1	Qualitative Specification using Constraint Preferences [SGAI'13]*	13
4.1.1	Syntax . . . . .	13
4.1.2	Semantics – Dominance Properties . . . . .	13
4.1.3	Transforming Constraint Preferences to Weighted Constraints . . . . .	13
4.1.4	Evaluation in the Energy Scenario [SEN-MAS'14]* . . . . .	13
4.2	Applications of Constraint Preferences . . . . .	13
4.2.1	Distributed Energy Management [SEN-MAS'14]* . . . . .	13
4.2.2	Multi-User Multi-Display Content Selection [ModRef'15] . . . . .	13
<b>5</b>	<b>Designing Algebraic Structures for Soft Constraints [Constraints'17]</b>	<b>15</b>
5.1	Abstracting Concrete Soft Constraint Formalisms – Requirements and Goals . . . . .	15
5.2	Relationship of Constraint Preferences with General Formalisms [ICTAI'14]* . . . . .	15
5.3	Building Algebraic Structures from Partial Orders . . . . .	15
5.3.1	Free Construction of Partial Valuation Structures over Partial Orders [Constraints'17]* . . . . .	15
5.3.2	Free Construction of C-Semirings over Partial Valuation Structures [Constraints'17]* . . . . .	15
5.4	Mapping Organizations – Combinations of Preference Structures [Wirsing'15]* . . . . .	15
5.4.1	Direct Product to Form Pareto-Orderings . . . . .	15
5.4.2	Lexicographic Product to Form Hierarchical Precedence . . . . .	15
<b>6</b>	<b>MiniBrass – A Modeling Language for Soft Constraints[Constraints'17]*</b>	<b>17</b>
6.1	Workflow . . . . .	18
6.2	Type Declarations . . . . .	18
6.3	PVS Types with Example Usages . . . . .	18
6.3.1	Comparative Specifications . . . . .	18
6.3.2	Cost-based Specifications . . . . .	18
6.3.3	Real-Valued Specifications . . . . .	18
6.4	Operators – Direct and Lexicographic Products . . . . .	18
6.5	A User-friendly Graphical Modeling Interface . . . . .	18
6.6	Consistency Checks and Utilities . . . . .	18
6.7	Soft Global Constraints . . . . .	18
6.8	PVS-based Generic Search . . . . .	18
6.8.1	Branch-and-Bound . . . . .	18
6.8.2	Large Neighborhood Search . . . . .	18
6.8.3	Adaptive Large Neighborhood Search . . . . .	18

<b>7</b>	<b>Evaluation</b>	<b>19</b>
7.1	Modeling Application Scenarios in MiniBrass . . . . .	19
7.1.1	Unit Commitment in Distributed Power Systems . . . . .	19
7.1.2	Task Allocation in Reconfigurable Swarms . . . . .	19
7.1.3	Self-organizing Resource-Flow Systems . . . . .	19
7.1.4	Multi-User Multi-Display Content Selection . . . . .	19
7.1.5	Mentor-Matching and Appointment Scheduling . . . . .	19
7.2	Benchmark Instances . . . . .	19
7.2.1	Comparing Performance: Encoded Weighted CSP versus Native Toulbar2 . . . . .	19
7.2.2	Comparing Models: Smyth-Optimization versus Weighted- Optimization . . . . .	19
7.2.3	Comparing Search Heuristics: Most Important First ver- sus Default . . . . .	19
7.2.4	Efficiency of Generic Heuristics . . . . .	19
<b>II</b>	<b>Self-organized Hierarchical Power Systems</b>	<b>21</b>
<b>8</b>	<b>Foundations</b>	<b>23</b>
8.1	Decentralized Power Management . . . . .	23
8.1.1	Current and Future Power Management Systems . . . . .	23
8.1.2	Smart Grids . . . . .	23
8.2	Mathematical Programming . . . . .	23
8.2.1	Linear Programming . . . . .	23
8.2.2	Mixed Integer Programming . . . . .	23
<b>9</b>	<b>Autonomous Virtual Power Plants to Decompose Distributed Unit Commitment [Vw. Steghöfer, Anders, Siefert]</b>	<b>25</b>
9.1	Hierarchies of Autonomous Virtual Power Plants [Vw. Anders] .	25
9.2	Trust To Manage Uncertainty [Vw. Anders] . . . . .	25
9.3	Soft Constraints in AVPPs [SEN-MAS'14]* . . . . .	25
<b>10</b>	<b>Supply Automata to Specify the Dynamics of Prosumers [TCCI'15]*</b>	<b>27</b>
10.1	Challenges in Unit-Commitment . . . . .	27
10.2	Existing Formulations . . . . .	27
10.3	Automata-based Specification of Optimization Problems . . . . .	27
10.3.1	Syntax . . . . .	27
10.3.2	Semantics . . . . .	27
10.3.3	Translation to Mixed Integer Programs . . . . .	27
10.4	Examples . . . . .	27
10.4.1	Modeling Cold and Hot Start-up . . . . .	27
10.4.2	Scheduling a Biogas-Plant . . . . .	27
10.5	Related Work . . . . .	27
<b>11</b>	<b>Abstraction of Collective Behavior [ICAART'14]* [TCCI'15]*</b>	<b>29</b>
11.1	Obtaining Simple Representations of Production Spaces . . . . .	29
11.2	Deriving Efficient Bounds for Future Time Steps . . . . .	29
11.3	Calculating Approximate Functions for Collectives . . . . .	29
11.3.1	Principle of Sampling Function Points . . . . .	29

11.3.2 Selection by means of Active Learning [SASO'15]* . . . .	29
11.4 Relationship with Unit-Commitment Algorithms . . . . .	29
11.4.1 Regio-Central Robust Optimization [SAOS'14] . . . . .	29
11.4.2 Market-Based Robust Optimization [TAAS'15] [SASO'15]	29
11.5 Related Work . . . . .	29
<b>12 Evaluation</b>	<b>31</b>
12.1 Efficiency of Abstractions in a Regio-Central Setting [ICAART'14]*	
[TCCI'15]* . . . . .	31
12.2 Efficiency of Abstractions in a Market-Based Setting [TAAS'15]*	31
12.3 Improving Accuracy of Sampling Abstraction by Active Learning	
[TAAS'15]* . . . . .	31
<b>13 Conclusion, Discussion, and Future Challenges</b>	<b>33</b>
13.1 Summary of Research Contributions . . . . .	33
13.2 Open Research Challenges . . . . .	33

\*

# Chapter 1

## Introduction

### 1.1 Motivation

### 1.2 Contributions of the Thesis



## Chapter 2

# Application Scenarios

### 2.1 Distributed Power Systems

#### 2.1.1 The Unit Commitment Problem

#### 2.1.2 Complexity of the Problem

#### 2.1.3 Related Work

### 2.2 Self-organizing Robotic Systems

#### 2.2.1 Task Allocation in Reconfigurable Swarms [ECAS'16]

#### 2.2.2 Self-organizing Resource-Flow Systems

#### 2.2.3 Related Work

### 2.3 Preference-oriented Adaptive Systems

#### 2.3.1 Exam Appointment Scheduling

#### 2.3.2 Mentor Matching [Constraints'17]\*

#### 2.3.3 Multi-User Multi-Display Exhibitions [ModRef'15]\*

#### 2.3.4 Related Work





## Part I

# Soft Constraints for Corridors of Optimal Behavior





## Chapter 3

# Foundations

### 3.1 Self-organization – Designing Systems to Adapt

### 3.2 Controlling Emergence – The Restore Invariant Approach

#### 3.2.1 Corridors of Correct Behavior

#### 3.2.2 Classical Constraint Programming for Combinatorial Problems

### 3.3 Over-Constrainedness in Practical Problems – From Correctness to Optimality

#### 3.3.1 Partial Constraint Satisfaction

#### 3.3.2 Specific Soft Constraint Formalisms

#### 3.3.3 Algebraic Structures for Soft Constraints

### 3.4 Algorithms to Solve Soft Constraint Problems

#### 3.4.1 Branch-and-Bound Search

#### 3.4.2 Large-Neighborhood Search

#### 3.4.3 Optimization in Constraint Programming Solvers

### 3.5 Modeling Languages

#### 3.5.1 MiniZinc and MiniSearch

#### 3.5.2 Essence

#### 3.5.3 Numberjack

## Chapter 4

# Soft Constraints with Constraint Preferences

### 4.1 Qualitative Specification using Constraint Preferences [SGAI'13]\*

#### 4.1.1 Syntax

#### 4.1.2 Semantics – Dominance Properties

#### 4.1.3 Transforming Constraint Preferences to Weighted Constraints

#### 4.1.4 Evaluation in the Energy Scenario [SEN-MAS'14]\*

### 4.2 Applications of Constraint Preferences

#### 4.2.1 Distributed Energy Management [SEN-MAS'14]\*

#### 4.2.2 Multi-User Multi-Display Content Selection [Mod-Ref'15]



## Chapter 5

# Designing Algebraic Structures for Soft Constraints [Constraints'17]

- 5.1 Abstracting Concrete Soft Constraint Formalisms – Requirements and Goals
- 5.2 Relationship of Constraint Preferences with General Formalisms [ICTAI'14]\*
- 5.3 Building Algebraic Structures from Partial Orders
  - 5.3.1 Free Construction of Partial Valuation Structures over Partial Orders [Constraints'17]\*
  - 5.3.2 Free Construction of C-Semirings over Partial Valuation Structures [Constraints'17]\*
- 5.4 Mapping Organizations – Combinations of Preference Structures [Wirsing'15]\*
  - 5.4.1 Direct Product to Form Pareto-Orderings
  - 5.4.2 Lexicographic Product to Form Hierarchical Precedence







## Chapter 6

# MiniBrass – A Modeling Language for Soft Constraints[Constraints’17]\*

### 6.1 Workflow

### 6.2 Type Declarations

### 6.3 PVS Types with Example Usages

#### 6.3.1 Comparative Specifications

The Free PVS

Constraint Preferences

#### 6.3.2 Cost-based Specifications

Max CSP

Weighted Constraints

Cost Function Networks

#### 6.3.3 Real-Valued Specifications

Fuzzy Constraints

Probabilistic Constraints

Possibilistic Constraints

### 6.4 Operators – Direct and Lexicographic Products

### 6.5 A User-friendly Graphical Modeling Interface

### 6.6 Consistency Checks and Utilities

### 6.7 Soft Global Constraints

### 6.8 PVS-based Generic Search

#### 6.8.1 Branch-and-Bound

## Chapter 7

# Evaluation

### 7.1 Modeling Application Scenarios in MiniBrass

#### 7.1.1 Unit Commitment in Distributed Power Systems

#### 7.1.2 Task Allocation in Reconfigurable Swarms

#### 7.1.3 Self-organizing Resource-Flow Systems

#### 7.1.4 Multi-User Multi-Display Content Selection

#### 7.1.5 Mentor-Matching and Appointment Scheduling

### 7.2 Benchmark Instances

#### 7.2.1 Comparing Performance: Encoded Weighted CSP versus Native Toulbar2

#### 7.2.2 Comparing Models: Smyth-Optimization versus Weighted- Optimization

#### 7.2.3 Comparing Search Heuristics: Most Important First versus Default

#### 7.2.4 Efficiency of Generic Heuristics



## Part II

# Self-organized Hierarchical Power Systems



## Chapter 8

# Foundations

### 8.1 Decentralized Power Management

#### 8.1.1 Current and Future Power Management Systems

#### 8.1.2 Smart Grids

### 8.2 Mathematical Programming

#### 8.2.1 Linear Programming

#### 8.2.2 Mixed Integer Programming





## Chapter 9

# Autonomous Virtual Power Plants to Decompose Distributed Unit Commitment [Vw. Steghöfer, Anders, Siefert]

- 9.1 Hierarchies of Autonomous Virtual Power  
Plants [Vw. Anders]
- 9.2 Trust To Manage Uncertainty [Vw. Anders]
- 9.3 Soft Constraints in AVPPs [SEN-MAS'14]\*



## Chapter 10

# Supply Automata to Specify the Dynamics of Prosumers [TCCI'15]\*

### 10.1 Challenges in Unit-Commitment

### 10.2 Existing Formulations

### 10.3 Automata-based Specification of Optimization Problems

#### 10.3.1 Syntax

#### 10.3.2 Semantics

#### 10.3.3 Translation to Mixed Integer Programs

### 10.4 Examples

#### 10.4.1 Modeling Cold and Hot Start-up

#### 10.4.2 Scheduling a Biogas-Plant

### 10.5 Related Work



## Chapter 11

# Abstraction of Collective Behavior [ICAART'14]\* [TCCI'15]\*

- 11.1 Obtaining Simple Representations of Production Spaces
- 11.2 Deriving Efficient Bounds for Future Time Steps
- 11.3 Calculating Approximate Functions for Collectives
  - 11.3.1 Principle of Sampling Function Points
  - 11.3.2 Selection by means of Active Learning [SASO'15']\*
- 11.4 Relationship with Unit-Commitment Algorithms
  - 11.4.1 Regio-Central Robust Optimization [SAOS'14]
  - 11.4.2 Market-Based Robust Optimization [TAAS'15] [SASO'15]
- 11.5 Related Work



## Chapter 12

# Evaluation

- 12.1 Efficiency of Abstractions in a Regio-Central Setting [ICAART'14]\* [TCCI'15]\*
- 12.2 Efficiency of Abstractions in a Market-Based Setting [TAAS'15]\*
- 12.3 Improving Accuracy of Sampling Abstraction by Active Learning [TAAS'15]\*





## Chapter 13

# Conclusion, Discussion, and Future Challenges

### 13.1 Summary of Research Contributions

### 13.2 Open Research Challenges