
INSTRUCTIONS FOR AUTHORS

HOW TO PREPARE PAPERS FOR LMCS USING `lmcs.cls` *

VERSION OF 2014-02-01

ALICE NAME1, BOB NAME2, AND CARLA NAME3

address 1

e-mail address: name1@email1

address2; addresses should initially be duplicated, even if authors share an affiliation

e-mail address: name2@email2; ditto for email addresses

address 3

URL: name3@url3 (optionally, a web-page can be specified)

ABSTRACT. The abstract has to precede the `maketitle` command. Be sure not to issue the `maketitle` command twice! In the abstract, mathematical expressions must be kept to the absolute minimum. Otherwise it should consist of plain ASCII text, without \TeX -commands, including explicit references using the `\cite` command. Presently we are not able to automatically extract an abstract containing such data and reliably turn it into html code. If you cannot meet these criteria, it is your responsibility to provide us with an html-version of your abstract. Please keep the abstract fairly short to prevent it from spilling over to the second page!

INTRODUCTION

Logical Methods in Computer Science is a community effort. It is run by scientists like you who devote their time and effort to make this a high-class open access journal that is free of cost for readers as well as authors. To minimize the extra work for the layout editor and to ensure smooth and fast publication of accepted articles, authors are asked to strictly adhere to the instructions for preparing their final version given in this document, which takes the form of a sample paper.

These revised instructions distill the experience of running the Journal for several years. They address the most time-consuming aspects of getting articles into publishable shape. While most articles require only minimal intervention, as the Journal's volume increased, so did the amount of time needed to reformat non-compliant articles.

Key words and phrases: MANDATORY list of keywords.

* OPTIONAL comment concerning the title, *e.g.*, if a variant or an extended abstract of the paper has appeared elsewhere.

thanks 2, optional.

thanks 3, optional.

TeX-nical matters. Please be aware that the class-file **lmcs.cls** supplied to authors will be replaced by the Journal's master class-file before publication of your article. Hence it is not necessary, and is in fact counterproductive, to emulate the appearance of published articles by means of your personal macros. Submissions not using **lmcs.cls** will be returned to the authors, as the reformatting that usually results from changing the class-file is usually too extensive and requires the original authors' intervention.

What authors *can* do to help the layout editor is to make their TeX- source compatible with the **hyperref**-package, which is included by the master class-file. In particular, care should be taken to use the `\texorpdfstring` macro for mathematical expressions in section or subsection headings (see the hyperref documentation for details: <https://ctan.org/pkg/hyperref>).

Authors must not (1) use unsupported fonts (like the **times**-package or the **txfonts**-package), (2) change the numbering style for theorems and definitions and the like, *e.g.*, by redefining the already provided proclamation environments for Theorems, Propositions, Lemmata, Corollaries etc. (you can add further environments, but those should comply with the default numbering style), and (3) use the `\sloppy` option globally. If it is impossible to achieve good line breaks by other means once the article is finished (reformulating a sentence, changing the word order, etc.), one can use `\sloppy` as a last resort *locally* in a paragraph.

Using lengthy mathematical expressions inside running text can lead to ugly breaks within formulae, even without producing overfull hboxes. If this is a persistent problem in your paper, please consider using more displayed formulae, or changing your notation.

The use of different macro-packages for the purpose of creating diagrams or other graphical displays is strongly discouraged. In the past that has led to papers that required different ways of processing to display the graphics of one type or the other, but could not easily be made to correctly display both types of graphics simultaneously. As a rule of thumb, as long as **pdflatex** correctly processes your paper, you should be in good shape. (Users of the **pstricks**-package and those used to including external eps-files should transform the resulting PostScript files to pdf.)

Please be aware that the proofs may display different vertical spacing in general, in particular different page breaks than the version originally submitted. If adjustments are deemed to be necessary by the authors, they can be implemented on the basis of the proofs, in collaboration with the layout editor, as a last step before final publication.

Matters of convenience. Please submit **only one file** containing the TeX-source of your paper! It is a major inconvenience when certain changes have to be applied in several files separately. Of course, we understand that separating a TeX source into several files has advantages during the creation of a paper, but please combine all parts into a single file for your submission. Your personal macros can of course be contained in a separate file, as can be external graphics. For the latter a dedicated subdirectory is required.

Matters of style. Your article should start with an introduction. This is the place to employ mathematical notation and give references, as opposed to the abstract. It is up to the authors to decide, whether to assign a section number to the introduction or not.

1. MULTIPLE AUTHORS

In papers with multiple authors several points need to be mentioned. Do not worry about footnote signs that will link author n to address n and the optional thanks n . This will be taken care of by the layout editor. Even if authors share an affiliation and part of an email address, they should follow the strict scheme outlined above and list their data individually. The layout editor will notice duplication of data and can then arrange for more space-efficient formatting. Alternatively, Authors can write “same data as Author n ” into some field to alert the layout editor. Unfortunately, so far we have not been able to devise a system that automatically takes care of these issues. But once the layout editor is made aware of some duplication, he can take some fairly simple measures to adjust the format accordingly. Placing the responsibility on the layout editor insures that these formatting issues are handled uniformly in different papers and that the authors do not have to second-guess the Journal’s policy.

2. USE OF DEFINITIONS AND THEOREMS ETC.

The numbering scheme for proclamations (Theorems, Definitions, etc.) uses the section number followed by the number of the current proclamation. There are no different “numbering threads” for the various types of proclamations, as then the relative position of, *e.g.*, Theorem 2.7 relative to, *e.g.*, Definition 2.9 would not be clear.

Definition 2.1. This is a definition.

Please use the supplied proclamation environments (as well as LaTeX’s cross-referencing facilities), or extend them in the spirit of the given ones, if necessary (*cf.* Satz 2.2 below). Refrain from replacing the Journal’s proclamation macros by your own constructs, especially do not change the numbering scheme: all proclamations are to be numbered consecutively!

2.1. First Subsection. This is a test of subsectioning. It works like numbering of paragraphs but is not linked with the numbering of theorems.

Satz 2.2. *This is a sample for a proclamation environment that can be added along with your personal macros, in case the supplied environments do not suffice. Please refrain from substituting other environments for the supplied ones. We distinguish those*

- *with italicised text:*
 - `\begin{thm}... \end{thm}` for a theorem;
 - `\begin{lem}... \end{lem}` for a lemma;
 - `\begin{cor}... \end{cor}` for a corollary;
 - `\begin{prop}... \end{prop}` for a proposition;
 - `\begin{asm}... \end{asm}` for an assumption;
- *and those with normal roman text:*
 - `\begin{defi}... \end{defi}` for a definition;
 - `\begin{conv}... \end{conv}` for a convention;
 - `\begin{conj}... \end{conj}` for a conjecture;
 - `\begin{fact}... \end{fact}` for facts;
 - `\begin{algo}... \end{algo}` for algorithms;
 - `\begin{pty}... \end{pty}` for a property;
 - `\begin{clm}... \end{clm}` for a claim;

- `\begin{nota}... \end{nota}` for a notation;
- `\begin{exa}... \end{exa}` for a single example;
- `\begin{exas}... \end{exas}` for a list of examples;
- `\begin{rem}... \end{rem}` for single remarks;
- `\begin{rems}... \end{rems}` for a list of remarks;
- `\begin{prob}... \end{prob}` for a single problem;
- `\begin{probs}... \end{probs}` for a list of problems;
- `\begin{oprob}... \end{oprob}` for a single open problem;
- `\begin{oprobs}... \end{oprobs}` for a list of open problems;
- `\begin{obs}... \end{obs}` for a single observation;
- `\begin{obss}... \end{obss}` for a list of observations;
- `\begin{qu}... \end{qu}` for a single question;
- `\begin{qus}... \end{qus}` for a list of questions.

The present new environment `\begin{satz}... \end{satz}` was defined by

`\theoremstyle{plain} \newtheorem{satz}[thm]{Satz}`

Proof. You can use the familiar `\begin{proof}... \end{proof}` construction. Please do not insert a blank line before `\end{proof}`, as this moves the box to a new line.

In case a proof ends with an itemization, please issue the command `\qedhere` at the end of the final item, *before* calling `\end{enumerate}` (or similar) and `\end{proof}`. Otherwise the end-of-proof box is put on a separate line following the last item, which looks awkward, unless the last line is too full to accommodate the box.

For options how to handle proofs ending in a displayed multi-line equation or formula, see this discussion. □

Corollary 2.3. *If no proof is given, `lmcs.cls` provides Paul Taylor’s end-of-proof box `\qed` to conclude a proclamation (Theorem, Proposition, Lemma, Corollary). Please do not redefine `\qed`!* □

3. ITEMIZATION

`lmcs.cls` provides the familiar environments

- (1) `\begin{itemize}... \end{itemize}` (see Satz 2.2 above)
- (2) `\begin{enumerate}... \end{enumerate}` (see this listing)
- (3) `\begin{description}... \end{description}`

in a form based on the `enumitem`-package, version 3.5.2 (please update, if you have an earlier version). This offers considerable simplifications, both for authors and the Layout Editor. Modifying the spacing of these environments is strongly discouraged. If you wish to change the labels, please consult the documentation of the `enumitem`-package. A simple example is found at the end of this document.

When proclamations or proofs start with an itemization without preceding text, two possibilities exist:

Theorem 3.1.

- (1) *Issuing an `\hfill`-command before the beginning of the list environment will push the first item to a new line, like in this case.*
- (2) *This is the second item.*

Proof.

- (1) The same behavior occurs in proofs; to start the first item on a new line an explicit `\hfill`-command is necessary.
- (2) The journal’s bibliography style is `alpha`, as exemplified by this citation [?]. Please use bibtex, i.e. supply a `.bib` file with the references in bibtex format along with your submission. □

We strongly recommend using this variant since it produces rather orderly output. The space-saving variant, in contrast, can look quite awful, *cf.* Theorem 3.2 below. Please notice that this paragraph is not indented, since it is following a proclamation that ended with a list environment. This can be achieved by starting the paragraph directly after the end of that environment, without inserting a blank line, or by explicit use of the `noindent`-command at the beginning of the paragraph. The effect indentation may have after a list environment is demonstrated after the proof of Theorem 3.2.

Theorem 3.2. (1) *Without the `\hfill`-command the first item starts in the same line as the title for the proclamation.*

- (2) *This may be useful when space needs to be conserved, but not in an electronic journal.*

Proof. (1) As you can see, the second option produces a somewhat unpleasant effect.

- (2) Hence we would urge authors to use the first variant. Perhaps a \TeX -guru can help us to make that the default, without the need for the `\hfill`-command. □

Here we started a new paragraph without suppressing its indentation. This adds to the rather disorienting appearance produced by not turning off the space-saving measures built into `amsart.cls`, on which this style is based. Please do issue the `\noindent` command in such situations, just as after the proof of Theorem 3.1 above.

ACKNOWLEDGMENT

The authors wish to acknowledge fruitful discussions with A and B.

REFERENCES

- [ABD⁺19] Giorgio Audrito, Jacob Beal, Ferruccio Damiani, Danilo Pianini, and Mirko Viroli. The share operator for field-based coordination. In *Lecture Notes in Computer Science*, pages 54–71. Springer International Publishing, 2019.
- [ABD⁺20] Giorgio Audrito, Jacob Beal, Ferruccio Damiani, Danilo Pianini, and Mirko Viroli. Field-based coordination with the share operator. *Logical Methods in Computer Science*, 2020. Submitted for publication.
- [APNF13] Bernhard Anzengruber, Danilo Pianini, Jussi Nieminen, and Alois Ferscha. Predicting social density in mass events to prevent crowd disasters. In *Social Informatics - 5th International Conference, SocInfo 2013, Kyoto, Japan, November 25-27, 2013, Proceedings*, pages 206–215, 2013.
- [AVD⁺19a] Giorgio Audrito, Mirko Viroli, Ferruccio Damiani, Danilo Pianini, and Jacob Beal. A higher-order calculus of computational fields. *ACM Transactions on Computational Logic*, 20(1):1–55, jan 2019.
- [AVD⁺19b] Giorgio Audrito, Mirko Viroli, Ferruccio Damiani, Danilo Pianini, and Jacob Beal. On a higher-order calculus of computational fields. In *Formal Techniques for Distributed Objects, Components, and Systems*, pages 289–292. Springer International Publishing, 2019.
- [BCC⁺20] Antonio Bucchiarone, Giacomo Cabri, Roberto Casadei, Mirko D’Angelo, Martina De Sanctis, Simon Dobson, Danilo Pianini, and Mirko Viroli. On social implications of collective adaptive systems. *IEEE Technology and Society Magazine*, 2020. Accepted for publication.

- [BPV15] Jacob Beal, Danilo Pianini, and Mirko Viroli. Aggregate programming for the internet of things. *IEEE Computer*, 48(9):22–30, 2015.
- [BVPD16] Jacob Beal, Mirko Viroli, Danilo Pianini, and Ferruccio Damiani. Self-adaptation to device distribution changes. In *10th IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2016, Augsburg, Germany, September 12-16, 2016*, pages 60–69, 2016.
- [BVPD17] Jacob Beal, Mirko Viroli, Danilo Pianini, and Ferruccio Damiani. Self-adaptation to device distribution in the internet of things. *ACM Trans. Auton. Adapt. Syst.*, 12(3):12:1–12:29, September 2017.
- [CFP⁺18] Roberto Casadei, Giancarlo Fortino, Danilo Pianini, Wilma Russo, Claudio Savaglio, and Mirko Viroli. Modelling and simulation of opportunistic IoT services with aggregate computing. *Future Generation Computer Systems*, sep 2018.
- [CFP⁺19] Roberto Casadei, Giancarlo Fortino, Danilo Pianini, Wilma Russo, Claudio Savaglio, and Mirko Viroli. A development approach for collective opportunistic edge-of-things services. *Information Sciences*, May 2019.
- [COPS19] Pierluigi Contucci, Andrea Omicini, Danilo Pianini, and Alina Sîrbu, editors. *The Future of Digital Democracy*. Springer International Publishing, 2019.
- [CPSV19] Roberto Casadei, Danilo Pianini, Guido Salvaneschi, and Mirko Viroli. On context-orientation in aggregate programming. In *2019 IEEE 4th International Workshops on Foundations and Applications of Self* Systems (FAS*W)*. IEEE, June 2019.
- [CPVN19] Roberto Casadei, Danilo Pianini, Mirko Viroli, and Antonio Natali. Self-organising coordination regions: A pattern for edge computing. In *Lecture Notes in Computer Science*, pages 182–199. Springer International Publishing, 2019.
- [CVA⁺19] Roberto Casadei, Mirko Viroli, Giorgio Audrito, Danilo Pianini, and Ferruccio Damiani. Aggregate processes in field calculus. In *Lecture Notes in Computer Science*, pages 200–217. Springer International Publishing, 2019.
- [CVA⁺20] Roberto Casadei, Mirko Viroli, Giorgio Audrito, Danilo Pianini, and Ferruccio Damiani. Engineering collective intelligence at the edge with aggregate processes. *Engineering Applications of Artificial Intelligence*, 2020. Submitted for publication.
- [DVFM⁺16] Simon Dobson, Mirko Viroli, Jose Luis Fernandez-Marquez, Franco Zambonelli, Graeme Stevenson, Giovanna Di Marzo Serugendo, Sara Montagna, Danilo Pianini, Juan Ye, Gabriella Castelli, and et al. Spatial awareness in pervasive ecosystems. *The Knowledge Engineering Review*, 31(4):343–366, Sep 2016.
- [DVPB15] Ferruccio Damiani, Mirko Viroli, Danilo Pianini, and Jacob Beal. Code mobility meets self-organisation: A higher-order calculus of computational fields. In *Formal Techniques for Distributed Objects, Components, and Systems - 35th IFIP WG 6.1 International Conference, FORTE 2015, Held as Part of the 10th International Federated Conference on Distributed Computing Techniques, DisCoTec 2015, Grenoble, France, June 2-4, 2015, Proceedings*, pages 113–128, 2015.
- [FPBV17] Matteo Francia, Danilo Pianini, Jacob Beal, and Mirko Viroli. Towards a foundational API for resilient distributed systems design. In *2nd IEEE International Workshops on Foundations and Applications of Self* Systems, FAS*W@SASO/ICCAC 2017, Tucson, AZ, USA, September 18-22, 2017*, pages 27–32, 2017.
- [MCF⁺19] Stefano Mariani, Roberto Casadei, Fabrizio Fornari, Giancarlo Fortino, Danilo Pianini, Barbara Re, Wilma Russo, Claudio Savaglio, Mirko Viroli, and Franco Zambonelli. Case studies for a new iot programming paradigm: Fluidware. In *Proceedings of the 1st Workshop on Artificial Intelligence and Internet of Things co-located with the 18th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2019), Rende (CS), Italy, November 22, 2019*, pages 82–96, 2019.
- [MOP15a] Sara Montagna, Andrea Omicini, and Danilo Pianini. Extending the gillespie’s stochastic simulation algorithm for integrating discrete-event and multi-agent based simulation. In *Multi-Agent-Based Simulation XVI - International Workshop, MABS 2015, Istanbul, Turkey, May 5, 2015, Revised Selected Papers*, pages 3–18, 2015.
- [MOP15b] Sara Montagna, Andrea Omicini, and Danilo Pianini. A gillespie-based computational model for integrating event-driven and multi-agent based simulation: Extended abstract. In *Proceedings*

- of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey, May 4-8, 2015, pages 1763–1764, 2015.
- [MPV12a] Sara Montagna, Danilo Pianini, and Mirko Viroli. Gradient-based self-organisation patterns of anticipative adaptation. In *Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2012, Lyon, France, September 10-14, 2012*, pages 169–174, 2012.
- [MPV12b] Sara Montagna, Danilo Pianini, and Mirko Viroli. A model for drosophila melanogaster development from a single cell to stripe pattern formation. In *Proceedings of the ACM Symposium on Applied Computing, SAC 2012, Riva, Trento, Italy, March 26-30, 2012*, pages 1406–1412, 2012.
- [MVR⁺11] Sara Montagna, Mirko Viroli, Matteo Risoldi, Danilo Pianini, and Giovanna Di Marzo Serugendo. Self-organising pervasive ecosystems: A crowd evacuation example. In *Software Engineering for Resilient Systems - Third International Workshop, SERENE 2011, Geneva, Switzerland, September 29-30, 2011. Proceedings*, pages 115–129, 2011.
- [PBV16] Danilo Pianini, Jacob Beal, and Mirko Viroli. Improving gossip dynamics through overlapping replicates. In *Coordination Models and Languages - 18th IFIP WG 6.1 International Conference, COORDINATION 2016, Held as Part of the 11th International Federated Conference on Distributed Computing Techniques, DisCoTec 2016, Heraklion, Crete, Greece, June 6-9, 2016, Proceedings*, pages 192–207, 2016.
- [PBV17] Danilo Pianini, Jacob Beal, and Mirko Viroli. Practical aggregate programming with protelis. In *2nd IEEE International Workshops on Foundations and Applications of Self* Systems, FAS*W@SASO/ICCAC 2017, Tucson, AZ, USA, September 18-22, 2017*, pages 391–392, 2017.
- [PCC⁺18] Danilo Pianini, Giovanni Ciatto, Roberto Casadei, Stefano Mariani, Mirko Viroli, and Andrea Omicini. Transparent protection of aggregate computations from byzantine behaviours via blockchain. In *Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good - Goodtechs 18*. ACM Press, 2018.
- [PCR15] Danilo Pianini, Angelo Croatti, Alessandro Ricci, and Mirko Viroli. Computational fields meet augmented reality: Perspectives and challenges. In *2015 IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASO Workshops 2015, Cambridge, MA, USA, September 21-25, 2015*, pages 80–85, 2015.
- [PCV19] Danilo Pianini, Roberto Casadei, and Mirko Viroli. Security in collective adaptive systems: A roadmap. In *2019 IEEE 4th International Workshops on Foundations and Applications of Self* Systems (FAS*W)*. IEEE, June 2019.
- [PCVN21] Danilo Pianini, Roberto Casadei, Mirko Viroli, and Antonio Natali. Partitioned integration and coordination via the self-organising coordination regions pattern. *Future Generation Computer Systems*, 114:44 – 68, 2021.
- [PDV17] Danilo Pianini, Simon Dobson, and Mirko Viroli. Self-stabilising target counting in wireless sensor networks using euler integration. In *11th IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2017, Tucson, AZ, USA, September 18-22, 2017*, pages 11–20, 2017.
- [PEGC18] Danilo Pianini, Ahmed Elzanaty, Andrea Giorgetti, and Marco Chiani. Emerging distributed programming paradigm for cyber-physical systems over LoRaWANs. In *2018 IEEE Globecom Workshops (GC Wkshps)*. IEEE, December 2018.
- [PMV11] Danilo Pianini, Sara Montagna, and Mirko Viroli. A chemical inspired simulation framework for pervasive services ecosystems. In *Federated Conference on Computer Science and Information Systems - FedCSIS 2011, Szczecin, Poland, 18-21 September 2011, Proceedings*, pages 667–674, 2011.
- [PMV13] Danilo Pianini, Sara Montagna, and Mirko Viroli. Chemical-oriented simulation of computational systems with ALCHEMIST. *J. Simulation*, 7(3):202–215, 2013.
- [PMVZ20] Danilo Pianini, Stefano Mariani, Mirko Viroli, and Franco Zambonelli. Time-fluid field-based coordination. In *Coordination Models and Languages - 22nd IFIP WG 6.1 International Conference, COORDINATION 2020, Held as Part of the 15th International Federated Conference on Distributed Computing Techniques, DisCoTec 2020, Valletta, Malta, June 15-19, 2020, Proceedings*, pages 193–210, 2020.
- [PO18] Danilo Pianini and Andrea Omicini. Democratic process and digital platforms: An engineering perspective. In *The Future of Digital Democracy*, pages 83–96. Springer International Publishing, dec 2018.

- [PPCE20] Federico Pettinari, Danilo Pianini, Roberto Casadei, and Lukas Esterle. A development and simulation framework for decentralised k-coverage in situated multi-robot systems. 2020. Ready to be submitted for publication.
- [PSV14] Danilo Pianini, Stefano Sebastio, and Andrea Vandin. Distributed statistical analysis of complex systems modeled through a chemical metaphor. In *International Conference on High Performance Computing & Simulation, HPCS 2014, Bologna, Italy, 21-25 July, 2014*, pages 416–423, 2014.
- [PVB15] Danilo Pianini, Mirko Viroli, and Jacob Beal. Protelis: practical aggregate programming. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing, Salamanca, Spain, April 13-17, 2015*, pages 1846–1853, 2015.
- [PVM⁺10] Danilo Pianini, Sascia Virruso, Ronaldo Menezes, Andrea Omicini, and Mirko Viroli. Self organization in coordination systems using a wordnet-based ontology. In *Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2010, Budapest, Hungary, 27 September - 1 October 2010*, pages 114–123, 2010.
- [PVM11] Danilo Pianini, Mirko Viroli, and Sara Montagna. A simulation framework for pervasive services ecosystems. In *Proceedings of the 12th Workshop on Objects and Agents, Rende (CS), Italy, Jul 4-6, 2011*, pages 150–157, 2011.
- [PVZF14] Danilo Pianini, Mirko Viroli, Franco Zambonelli, and Alois Ferscha. HPC from a self-organisation perspective: The case of crowd steering at the urban scale. In *International Conference on High Performance Computing & Simulation, HPCS 2014, Bologna, Italy, 21-25 July, 2014*, pages 460–467, 2014.
- [RVO⁺17] Alessandro Ricci, Mirko Viroli, Andrea Omicini, Stefano Mariani, Angelo Croatti, and Danilo Pianini. *Spatial Tuples: Augmenting Physical Reality with Tuple Spaces*, pages 121–130. Springer International Publishing, Cham, 2017.
- [RVO⁺18] Alessandro Ricci, Mirko Viroli, Andrea Omicini, Stefano Mariani, Angelo Croatti, and Danilo Pianini. Spatial tuples: Augmenting reality with tuples. *Expert Systems*, page e12273, apr 2018.
- [SYD⁺13] Graeme Stevenson, Juan Ye, Simon Dobson, Danilo Pianini, Sara Montagna, and Mirko Viroli. Combining self-organisation, context-awareness and semantic reasoning: the case of resource discovery in opportunistic networks. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing, SAC '13, Coimbra, Portugal, March 18-22, 2013*, pages 1369–1376, 2013.
- [VAB⁺18] Mirko Viroli, Giorgio Audrito, Jacob Beal, Ferruccio Damiani, and Danilo Pianini. Engineering resilient collective adaptive systems by self-stabilisation. *ACM Transactions on Modeling and Computer Simulation*, 28(2):1–28, mar 2018.
- [VBD⁺18] Mirko Viroli, Jacob Beal, Ferruccio Damiani, Giorgio Audrito, Roberto Casadei, and Danilo Pianini. From field-based coordination to aggregate computing. In *Lecture Notes in Computer Science*, pages 252–279. Springer International Publishing, 2018.
- [VBD⁺19] Mirko Viroli, Jacob Beal, Ferruccio Damiani, Giorgio Audrito, Roberto Casadei, and Danilo Pianini. From distributed coordination to field calculus and aggregate computing. *Journal of Logical and Algebraic Methods in Programming*, 109:100486, December 2019.
- [VBDB15] Mirko Viroli, Jacob Beal, Ferruccio Damiani, and Danilo Pianini. Efficient engineering of complex self-organising systems by self-stabilising fields. In *2015 IEEE 9th International Conference on Self-Adaptive and Self-Organizing Systems, Cambridge, MA, USA, September 21-25, 2015*, pages 81–90, 2015.
- [VBPB16] Mirko Viroli, Antonio Bucchiarone, Danilo Pianini, and Jacob Beal. Combining self-organisation and autonomic computing in CASs with aggregate-MAPE. In *2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS*W)*. IEEE, sep 2016.
- [VCP16a] Mirko Viroli, Roberto Casadei, and Danilo Pianini. On execution platforms for large-scale aggregate computing. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp Adjunct 2016, Heidelberg, Germany, September 12-16, 2016*, pages 1321–1326, 2016.
- [VCP16b] Mirko Viroli, Roberto Casadei, and Danilo Pianini. Simulating large-scale aggregate mass with alchemist and scala. In *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, pages 1495–1504, 2016.

- [VPB12] Mirko Viroli, Danilo Pianini, and Jacob Beal. Linda in space-time: An adaptive coordination model for mobile ad-hoc environments. In *Coordination Models and Languages - 14th International Conference, COORDINATION 2012, Stockholm, Sweden, June 14-15, 2012. Proceedings*, pages 212–229, 2012.
- [VPM⁺15] Mirko Viroli, Danilo Pianini, Sara Montagna, Graeme Stevenson, and Franco Zambonelli. A coordination model of pervasive service ecosystems. *Sci. Comput. Program.*, 110:3–22, 2015.
- [VPMS12] Mirko Viroli, Danilo Pianini, Sara Montagna, and Graeme Stevenson. Pervasive ecosystems: a coordination model based on semantic chemistry. In *Proceedings of the ACM Symposium on Applied Computing, SAC 2012, Riva, Trento, Italy, March 26-30, 2012*, pages 295–302, 2012.
- [VPR⁺15] Mirko Viroli, Danilo Pianini, Alessandro Ricci, Pietro Brunetti, and Angelo Croatti. Multi-agent systems meet aggregate programming: Towards a notion of aggregate plan. In *PRIMA 2015: Principles and Practice of Multi-Agent Systems - 18th International Conference, Bertinoro, Italy, October 26-30, 2015, Proceedings*, pages 49–64, 2015.
- [VPRC17] Mirko Viroli, Danilo Pianini, Alessandro Ricci, and Angelo Croatti. Aggregate plans for multiagent systems. *International Journal of Agent-Oriented Software Engineering*, 5(4):336, 2017.
- [ZOA⁺15] Franco Zambonelli, Andrea Omicini, Bernhard Anzenberger, Gabriella Castelli, Francesco L. De Angelis, Giovanna Di Marzo Serugendo, Simon A. Dobson, Jose Luis Fernandez-Marquez, Alois Ferscha, Marco Mamei, Stefano Mariani, Ambra Molesini, Sara Montagna, Jussi Nieminen, Danilo Pianini, Matteo Risoldi, Alberto Rosi, Graeme Stevenson, Mirko Viroli, and Juan Ye. Developing pervasive multi-agent systems with nature-inspired coordination. *Pervasive and Mobile Computing*, 17:236–252, 2015.

APPENDIX A.

Here is a check-list to be completed before submitting the paper to LMCS:

- ▷ your submission includes the latest version of `lmcs.cls`, that is, it does not rely on the version of the class file provided by arXiv
- ▷ the text of your submission is contained in a single file, except for macros and graphics
- ▷ your graphics use only one format
- ▷ you have employed the Journal’s original proclamation environments, or suitable extensions thereof
- ▷ you have loaded the `hyperref` package
- ▷ you have *not* loaded the `times` package
- ▷ you have not routinely adjusted vertical spacing manually by issuing `\vspace` or `\vskip` commands
- ▷ you have used the command `\sloppy` only locally and in emergency cases
- ▷ your displayed equations use the `\[...\]` construct
- ▷ your abstract only contains as few math-expressions as possible and no references
- ▷ your references are supplied in bibtex format in a separate `.bib` file

This listing also shows how to override the default bullet `•` of the `itemize`-environment by a different symbol, in this case `\triangleright`.