

# Towards Software Support for Collaborative Morphological Analysis

Marin Zec

Lehrstuhl für Software Engineering betrieblicher Informationssysteme,  
Technische Universität München

## Abstract

Today's knowledge workers regularly face ill-structured, non-routine wicked problems such as strategy development, product innovation or policy design. Morphological analysis (MA) is a versatile problem structuring method that helps teams to build a shared understanding of an issue, generate solution ideas and support decision-making. However, existing MA software has two major drawbacks given the high prevalence of teamwork in modern organizations: (1) it only provides a single user interface and (2) it neglects established practices to improve group performance and creativity because there is no collaboration support. I propose extensions of the classical process model for morphological analysis to incorporate findings from cognitive and social psychology that may serve as a basis for collaborative MA software implementations and discuss the underlying design decisions.

## 1 Morphological Analysis

Morphological analysis (MA) is a method to analyze and structure complex problems as well as idea generation. MA has been used widely both in industry and academia. It involves three major stages: (1) analysis, (2) synthesis, and (3) exploration. The core idea of MA is to approach the problem at hand by studying its formal structure (morphology): the problem is modeled as a system composed of subsystems and a solution candidate is conceived as a combination of solutions to the subproblems. The result of the decomposition (i.e. the problem space) is captured in a morphological matrix (MM; also morphological box or field). MA can be conducted both physically using pen-and-paper/whiteboard as well as digitally using software. However, the synthesis and exploration stage benefit significantly from software support (particularly for larger MMs). A major obstacle to adoption of MA (software) by (virtual) teams is the lack of software support for *collaborative* MA.

## 1.1 Analysis Stage

Within the analysis stage, the problem is decomposed into subproblems (*parameters*) and for each subproblem a range of possible solutions (*values*) is generated. Then, solution candidates (*configurations*) are generated by selecting one value per parameter. At this stage, creative and contingent thinking is desirable to make the set of potential solutions as large as possible. Finally, the MM is created: a compact matrix representation of the formal solution space containing all formal solution candidates (see figure 1).

## 1.2 Synthesis Stage

The purpose of the synthesis stage is to filter out infeasible configurations. Non-scholarly literature on problem solving and creativity techniques typically introduces a shortened account of MA that focuses on the analysis phase and provides little if any structured recommendations for synthesizing potential solutions. A major reason for the popularity of this simplified version of MA might be the lack of access to easy-to-use software. However, MA software provides some significant advantages: it can efficiently iterate over all formal solution candidates and filter out infeasible solution candidates (as opposed to time-consuming manual configuration evaluations in physical settings) based on a cross consistency matrix (CCM; see figure 2) specified by the user(s). The CCM contains all the consistency assessments of each pair of parameter values.

Parameter A	Parameter B	Parameter C
Value A1	Value B1	Value C1
Value A2	Value B2	Value C2
Value A3	Value B3	Value C3
	Value B4	

Figure 1: A morphological matrix consisting of three parameters. The matrix represents 36 configurations such as (A2, B3, C1).

		A			B				C		
		A1	A2	A3	B1	B2	B3	B4	C1	C2	C3
B	B1	X									
	B2	X									
	B3			X							
	B4		X								
C	C1				X						
	C2	X									
	C3			X	X		X				

Figure 2: A (symmetric) cross consistency matrix ("X" represents inconsistency).

## 1.3 Exploration Stage

A major advantage of MA software over manual approaches is (interactive) visualization support. Based on the CCM, MA software can calculate the maximum subset of consistent configurations even for a large set of formal solution configurations. This is typically nonviable in a pen and paper workshop setting. In addition, MA software provides various kinds of (interactive) representations of the consistent configuration space. Such representations help groups to structure discussion and support decision-making (see figures 3 and 4). For instance, MA/Carma facilitates exploratory what-if analyses by specifying fixed or exogenous values (red). Remaining parameter values that preserve consistency are highlighted (blue). Parmenides/EIDOS maps the multidimensional configuration space to a two-dimensional plane for visual cluster analysis. Each circle represents one configuration. Spatial proximity indicates the degree of similarity between configurations.

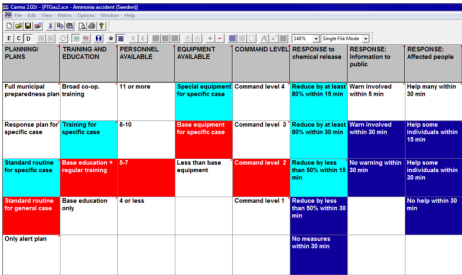


Figure 3: MA/Carma provides an interactive inference model<sup>1</sup>.

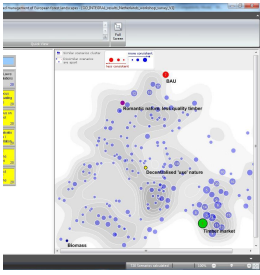


Figure 4: Parmenides Eidos provides means for visual cluster analysis<sup>2</sup>.

## 2 Collaborative Morphological Analysis

The possibility to filter out inconsistent configurations and provide (interactive) visualizations is a strong argument in favor of using software to conduct MA. However, existing MA software does have two major drawbacks. First, it does not provide a multi-user interface, which renders it impracticable for use in distributed teams. Second, results from CSCW and psychology literature did not receive any consideration in their process model design. A large body of research shows that groups are prone to process losses such as production blocking, social loafing, social inhibition, group polarization, hidden or the common knowledge effect. To increase the adoption and value of MA for teams, the classical process model has to be adapted to support teams in avoiding process losses. In addition, implementations of such a collaborative MA (CMA) approach should acknowledge and take into account that team members increasingly collaborate across temporal, spatial and organizational boundaries.

<b>Hypothesis 1</b>	<i>The lack of implementations of (C)MA as a multi-user interface application is the primary barrier to adoption of MA for (virtual) teams.</i>
<b>Hypothesis 2</b>	<i>Explicit iterative loops between private individual and public collaborative sub-steps in the analysis and synthesis stages of MA increase the quality and richness of the MM and CCM, respectively, by mitigating negative effects of anchoring, social inhibition, group polarization, and the common knowledge effect.</i>
<b>Hypothesis 3</b>	<i>Software-supported model merging and interactive conflict resolution mechanisms decrease the MA process duration and help mitigate the common knowledge effect by setting the primary focus of the discussion on controversial issues (based on automated matrix comparisons).</i>

Table 1: The major hypotheses underlying the proposed CMA process and software support.

<sup>1</sup> Source: <http://www.swemorph.com/macarma.html>. Accessed 8 July 2015.

<sup>2</sup> Source: [https://forestwiki.jrc.ec.europa.eu/integral/index.php/South-East\\_Veluwe](https://forestwiki.jrc.ec.europa.eu/integral/index.php/South-East_Veluwe). Accessed 8 July 2015.

Based on the hypotheses derived from literature (see table 1), I propose the subdivision of the analysis and synthesis stage into two iterative substeps (Zec et al. 2015): an individual and a collaborative step. Multiple iterations are allowed to enable cognitive stimulation through idea exchange. While divergent and contingency thinking is desirable at the beginning of each phase, the individual matrices (MMs and CCMs, respectively) have to converge into a shared matrix at the end of each stage that will serve as the starting point for the next stage. Therefore, model merging and conflict resolution mechanisms have to be provided to facilitate progress. Figure 5 shows an overview of the proposed CMA process model.

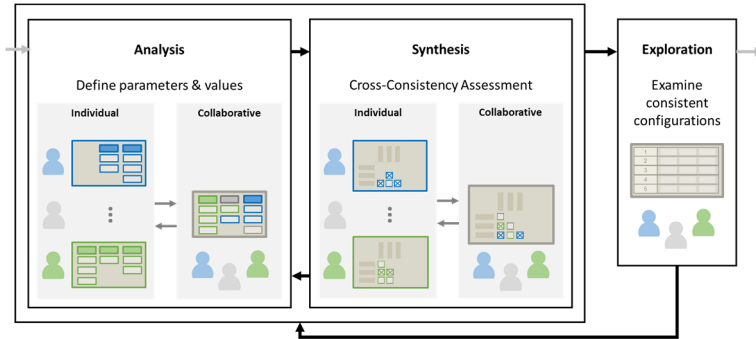


Figure 5: The proposed process model for CMA.

### 3 Conclusions and Future Work

The lack of multi-user interfaces and adequate collaborative process models are barriers to adoption of available software in MA. In this work, I derived major hypotheses that are supposed to guide the design of a CMA process model and respective software implementations. The next step is to validate the postulated hypotheses empirically and refine the CMA process model as well as the prototypical implementation accordingly.

#### Acknowledgments

This research was sponsored in part by the German Federal Ministry of Education and Research (BMBF) (project “Software Campus (TU München)”, grant no. 01IS12057).

#### Contact Information

Marin Zec, M.Sc.  
Lehrstuhl für Software Engineering betrieblicher Informationssysteme, TU München  
Boltzmannstraße 3, 85748 Garching  
marin.zec@tum.de

#### References

Zec, M., Schneider, A. W., & Matthes F. (2015). Towards a Process Model for Computer-Supported Collaborative Morphological Analysis. In *AMCIS 2015 Proceedings*.