

Complexity in Analogy Tasks: An Analysis and Computational Model

Philip Stahl and Marco Ragni¹

Abstract. In this article, we introduce a complexity measure for matrix tasks which occur in IQ-tests with respect to the kind of functions necessary to solve such tasks. The aim is to capture human reasoning difficulty. We implemented a program which is able to solve matrix tasks and to evaluate their complexity by our measure. The results of the evaluation are compared with the empirical difficulty ranking from Cattell's Culture Fair Test.

1 Functional analysis of matrix tasks

For about a hundred years intelligence has mostly been tested using IQ-tests like Cattell's Culture Fair Test (CFT) [3]. IQ-tests measure the intelligence of an individual by determining the deviation from a group (of age and educational status, etc.) on a given set of reasoning tasks. Tasks are defined as simple or hard if (empirically) a given population is able to solve most or a limited number of such tasks.

While it is possible to empirically capture human reasoning difficulty – it is also possible to analyze such tasks formally. Consequently applying the idea of Newell and Simon [2] of cognition as computation, we can classify tasks with respect to their computational requirements. We implemented a program which solves matrix tasks (see fig. 1 for an example task, fig. 2 for the GUI) by analyzing

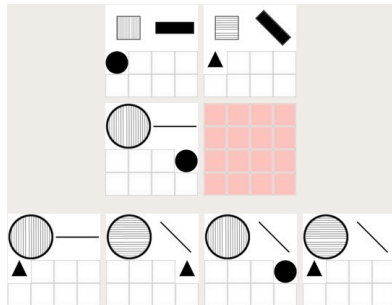


Figure 1. A 2x2 matrix task. The lower four fields are possible answers.

ing their underlying functions. We introduce a complexity measure which allows the program to give a detailed evaluation of the complexity of each solved task. We put this approach to the test for one of the best analyzed sample sets for the CFT which offers an empirical difficulty rating. Our measure is evaluated against this sample set. We analyzed the tasks of the CFT according to their underlying

structure, the stimuli and the occurring transformations to understand basic principles of these tasks.

The CFT contains four different subtests. The tasks of subtest 1 (row completion) and subtest 3 (matrix tasks) can be seen as 1x4, 2x2, or 3x3 matrices. The lower right field ([1,4],[2,2],[3,3]) of each matrix is always empty. The task is to select the matching answer field from four given possibilities. Therefore it is necessary to identify a certain regularity between the stimuli in the single fields.

The regularity can be described as a set of relations between the stimuli in the different fields. The relations go beyond the pure visual information of the stimuli and have to be derived from the comparison of properties of the different stimuli. The differences between stimuli can be described by functions, which transfer one stimulus into another. The less functions are required to determine the stimuli in the answer field, the more easy the task can be solved.

We give now a short survey over the implementation of our program, which runs for CFT-like tasks. First, we describe how the properties of each stimulus and the relations between several stimuli in a matrix field are represented. Then the association between stimuli of different matrix fields are defined.

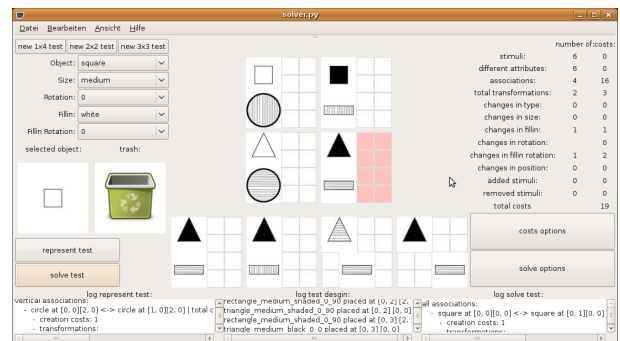


Figure 2. The GUI of our program. Left are properties defining a stimulus. On the right is the output of the analysis for all identified transformations (applied and within the associations).

Before a task can be solved, the stimuli itself have to be represented. Therefore a hierarchy of properties must be specified which determines the importance of a single property. The hierarchy must consider that some properties of a stimulus are more obvious for a reasoner than others. The hierarchy of properties allows to generate a hierarchy between the stimuli in one field, such that lower ranked stimuli can be related to higher ranked stimuli. The higher ranked stimulus works like a reference, which allows one to formulate relations like 'left to the big triangle', 'below the big triangle', etc.

¹ University of Freiburg, email: philip.stahl@mars.uni-freiburg.de, ragni@cognition.uni-freiburg.de

The second important point for solving a matrix task is the correct association building between the stimuli of the different matrix fields. An association creates a relationship between two or three single fields by identifying a stimulus in each one which represents the state of an object which occurs in all of these fields. The association represents the fact that some stimuli in different fields are perceived as one object, if they share a basic similarity according to the RD theory [1]. The state of the object may vary in some properties, but some remain equal as a certain indicator which distinguishes the stimulus from other stimuli. These characteristics can vary from task to task according to the different relations between stimuli.

The *complexity of an association* depends on the structure of the task: The same association can have a different complexity in two different tasks, because of the number of stimuli which have to be compared to identify the association. Many similar stimuli lead to a difficult association building, whereas fewer different stimuli lead to an easy building. Correct association building is a basic procedure for solving a matrix task. Associations allow the program to identify transformations and relations between the stimuli in the different matrix fields. Also, it can be used to make a prediction about the stimuli which should occur in the answer field.

Different task type require different built associations to specify the right answer. Mainly two different ways can be distinguished: Row completion is chiefly used for 1x4 and 3x3 matrix tasks, whereas transformation-transfer solves 2x2 tasks.

Row completion uses the fact that a stimulus occurring in a field which lies in a line or row leading to the answer field, also occurs in the answer field. (Except it is removed by a transformation.) If an association exists over the fields [1,1],[1,2],[1,3] in a 1x4 task or over fields [3,1], [3,2] (or [1,3], [2,3]) in a 3x3 task, it can be predicted that the stimulus of the association occurs in the answer field. Its properties can be computed by applying the transformations of the association on the stimulus in the previous field to the answer field.

The row completion procedure can not be applied for 2x2 tasks. These tasks require a shift of transformations from a figure A-B to a figure C-ANSWER. They are solved in the following way: The same transformations which transfer field A into field B, also transfer figure C into the answer figure. Due to the fact that a 2x2 task has a horizontal and vertical dimension, a single figure is not always identical to a single field. The figures have to be identified first by associating the right stimuli.

In brief, we described two algorithms which are able to solve 1x4, 3x3 and 2x2 matrix tasks. Both algorithms work on the association building between the single stimuli and the application of transformations. Now we develop a cost measure which evaluates the complexity of a task by its number of associations and applied transformations. We compare it to the empirical findings of the CFT.

2 Complexity measure

The costs of a single task are computed by the number of associations and transformations which are required to solve a task.

Since some kinds of transformations are easier for humans to apply than others we introduced weights for each different transformation type, which should simulate their complexity for humans. To specify the weights we made a detailed analysis of the cost measure given by the CFT. Consequently, it is possible to explain the difficulty of each task in the CFT by analyzing its stimuli and transformations. We made the following assumptions: Scaling (s), translating a stimulus (m), changing its fillin (c) or adding and removing it (ar) seem to be similar in their complexity and amount to the basic costs of 1 unit.

Rotation appears to be more difficult, since the rotation of the filing of a stimulus (rf, costs 3 units) is even harder than the rotation of the stimulus itself (r, costs 2 units). Most difficult seems to be the transformation of the form/type (f) of a stimulus e.g. transferring a triangle into a square (costs 4 units).

The costs of all applied transformations which are required to specify the properties of the stimuli in the answer field are the sum of the single applied transformations multiplied by their weights:

$$C_{transform} = \sum_{t \in \{m,s,c,ar,r,f\}} (\#t \cdot w_t)$$

There are two components determining the costs of a built association. First there is the number of comparisons needed to build the association, reflecting the influence of other stimuli to the association. If there are many similar stimuli in each field it becomes harder to identify an association than if there are only a few ones, which are dissimilar to each other. Second, the number of transformations applied between the associated stimuli is regarded. The more unlike the associated stimuli are, the more difficult it becomes to associate them with each other, and the higher the costs rise. $C_{association} = \#comparisons + C_{transform}(association)$ The costs of the whole task are computed by summing up the costs of all built associations and the costs of all applied transformations multiplied with their weights.

$$C_{total}(task) = \sum_{a \in associations} (C_{association}(a)) + C_{transform}$$

We compared our cost measure to the CFT which evaluates a task regarding its complexity for a human solver.

The empirical classification of the CFT tasks (selected) can be seen in Table 1. The value P of the CFT reflects the percentage of people able to solve this task. The correlation for our complexity measure with the empirical data is significant: with weight $r = -0.74, p = 0.015$ (all weights according to the values above) and without weight $r = 0.72, p = 0.02$ (transformation costs 1 unit).

Table 1. Analysis of the cost function for tasks in CFT3. Index P = the empirical difficulty ranking, C = complexity with weight and without

TASK	Index P	C(Weight)	C()
CFT3-A-1-3-1	76	8	7
CFT3-A-2-1-1	96	5	3
CFT3-A-2-1-3	95	6	3
CFT3-A-2-1-4	90	4	4
CFT3-A-2-3-2	94	10	6
CFT3-A-2-3-5	47	26	18
CFT3-A-2-3-6	85	21	17

The presented cognitive complexity measure for matrix tasks allows us to classify tasks with respect to the number of basic functions they need to be solved. This measure is based on abstract 'units' and reflects partially cognitive complexity due to the high and significant correlations with empirical data.

REFERENCES

- [1] N. Chater and U. Hahn, 'Representational distortion, similarity and the universal law of generalization', in *Proceedings of the similarity and categorization workshop* 97, pp. 31–36, (1997).
- [2] A. Newell and H. A. Simon, *Human problem solving*, Prentice-Hall, Englewood Cliffs, NJ, 1972.
- [3] R. H. Weiss, *Grundintelligenztest Skala 3 - CFT 3*, Stuttgart, 1971.