Objective measures of complexity for dynamic decision making in an interactive learning environment

Sylvain Pronovost, Jean-François Gagnon, Daniel Lafond, Sébastien Tremblay

Co-DOT laboratory, School of Psychology, 2325, rue des Bibliothèques, Québec, QC G1K 7P4 Canada
Thales Research & Technology Canada, 1405, Parc Technologique, Québec, QC G1P 4P5 Canada

sylvain.pronovost@psy.ulaval.ca, jean-francois.gagnon.22@ulaval.ca,
Daniel.Lafond@ca.thalesgroup.com, Sebastien.Tremblay@psy.ulaval.ca

APPENDIX A – OVERVIEW OF THE OBJECTIVE MEASURES OF COMPLEXITY

This section presents the seven objective models of complexity in their detailed calculations. A number of amendments were made in order to adapt those models from the domains of algorithmic and systemic complexity theory to dynamic decision making.

Cyclomatic Complexity (CC)

McCabe's Cyclomatic Complexity (1976), is a measure inspired by algorithmic graph theory, which determines the number of independent paths in a control flow graph. It is computed as a simple subtraction of the number of graph nodes from graph edges, plus the number of connected components.

The **cyclomatic number** *V(G)* of a graph *G* with *n* vertices, *e* edges, and *p* connected components is:

$$v(G) = e - n + 2p$$

Halstead's Software Metrics (HM)

We use a modified version of Halstead's Software Metrics (1977), an implementation-independent complexity measure of algorithms. Halstead's metric depends on the number of unique and total operators and operands, and is thus a measure of algorithmic size, a strictly informational measure of complexity**.**

Halstead's measures are focused at symbolic complexity, and encompass a number of derived computations:

$n1$ = number of unique operators (*if*, *while*, etc.)
$n2$ = number of unique operands (variables or constants)
$N1$ = total number of operator occurrences
$N2$ = total number of operand occurrences

Vocabulary: $n = n1+n2$
Length: $N = N1+N2$
Calculated program length: $\hat{N} = n1*log2(n1) + n2*log2(n2)$
Volume: $V = (N1+N2)*log2(n1+n2)$
Difficulty: $D = (n1/2)*(N2/n2)$

Interface Complexity (IC)
Cardoso's Interface Complexity (Cardoso et al, 2006), adapted from Henry and Kafura's Information Flow (1981), determines the total complexity of an algorithm as the product of structural complexity and information flow, i.e., inputs and outputs**.**

Calculating the fan-in: number of local flows into that procedure plus the number of data structures from which that procedure retrieves information.
Calculating the fan-out: number of local flows out of that procedure plus the number of data structures that the procedure updates.

The complexity value is procedure length multiplied by the square of fan-in multiplied by fan-out:

$$(length \times (fan\text{-}in \times fan\text{-}out)^2)$$

Adaptation of the measure by Cardoso, Mendling, Neumann & Reijers (2006), for Business Process Model and Notation (BPMN):

Interface Complexity (*IC*) of an activity:

$$IC = length * (number\ of\ inputs * number\ of\ outputs)^2$$

Cognitive Functional Size (CFS) [1]
Wang's Cognitive Functional Size (Shao and Wang, 2003) computes the complexity of an algorithm as a function of the product of summed inputs and outputs by the "total cognitive weight", this latter construct being itself an additive function of "basic control structures" which bear various cognitive weights determined via empirical studies**.**

The *cognitive functional size* of a basic algorithm, S_f :

$$S_f = N_{i/o} \times W_c = (N_i + N_o) \cdot \{\sum_{j=1}^{q} [\prod_{k=1}^{m} \sum_{i=1}^{n} w_c(j,k,i)]\} \text{ [CWU]},$$

is defined as a product of the sum of inputs and outputs ($N_{i/o}$), and the total cognitive weight:

$$W_c = \sum_{j=1}^{q} [\prod_{k=1}^{m} \sum_{i=1}^{n} w_c(j,k,i)].$$

[1] It should be noted that Shao and Wang's (2003) CFS used a different set of basic control structures than the $C_c(S)$, as well as a different metric to compute them, which was reused in Misra's CWCM (2006) and Kushwaha and Misra's CICM (2006).

The unit of cognitive weight being sequence with 1 *i/o* :

$$\begin{aligned} S_{f0} &= f(N_{i/o}, W_{bcs}) \\ &= (N_i + N_o) \cdot W_c \\ &= 1 \times 1 \\ &= 1 \ \text{[CWU]}, \end{aligned}$$

Cognitive Weight Complexity Measure CWCM)
Misra's Cognitive Weight Complexity Measure (2006), based on Wang's CFS, is a metric considering exclusively the total of cognitive weights, as a function of executed instructions in an algorithm**.**

$$CWCM = Wc$$

where *Wc* is the total cognitive weight of software (see the calculation of the CFS model for details).

Cognitive Information Complexity Measure (CICM)
Kushwaha and Misra's Cognitive Information Complexity Measure (2006), also based on Wang's CFS, is a metric combining a weighted information score multiplied by the total cognitive weight.

$$CICM = WICS * Wc$$

where *WlCS* is the sum of weighted information count of all the lines in an algorithm (see the calculation of the CFS model for details).

Cognitive Complexity (CcS)
Wang's Cognitive Complexity (Wang, 2007, 2009) is a more objective and rigorous measure of a system's complexity and size, because it represents its real *semantic complexity* (as opposed to mere symbolic quantification) by integrating both the *operational complexity* and the *architectural complexity* of a system in a coherent measure.

The *operational complexity* of a system *S*, $C_{op}(S)$, is determined by the sum of the cognitive weights of its *n* linear blocks composed by individual BCS's, where each block may consist of *q* layers of embedded BCS's, and within each of the layer there are m linear BCS's:

$$\begin{aligned} C_{op}(S) &= \sum_{k=1}^{n_c} C_{op}(C_k) \\ &= \sum_{k=1}^{n_c} \left(\prod_{j=1}^{q_k} \sum_{i=1}^{m_{k,j}} w(k,j,i)\right) \ \text{[F]} \end{aligned}$$

The *architectural complexity* of a system *S*, $C_a(S)$, is determined by the number of data objects at system and component levels:

$$\begin{aligned} C_a(S) &= \text{OBJ}(S)) \\ &= \sum_{k=1}^{n_{CLM}} \text{OBJ}(CLM_k) + \sum_{k=1}^{n_C} \text{OBJ}(C_k) \ \text{[O]} \end{aligned}$$

Where *OBJ* is a function that counts the number of data objects in a given *Component Logical Model* (CLM),which is equivalent to the number of global variables or components (number of local variables).

The *cognitive complexity* $C_c(S)$ of a system *S* is a product of the *operational complexity* $C_{op}(S)$ and the *architectural complexity* $C_a(S)$, that is:

$$\begin{aligned} C_c(S) &= C_{op}(S) \bullet C_a(S) \\ &= \{\sum_{k=1}^{n_C} \sum_{i=1}^{\#(C_s(C_k))} w(k,i)\} \bullet \\ &\quad \{\sum_{k=1}^{n_{CLM}} \mathrm{OBJ}(CLM_k) + \sum_{k=1}^{n_C} \mathrm{OBJ}(C_k)\} \quad [\mathrm{FO}] \end{aligned}$$

To summarize, cognitive complexity is:

$$C_c(S) = C_{op}(S) * C_a(S)$$

Where operational complexity $C_{op}(S)$ is the total cognitive weight using revised calibrated cognitive weights for BCSs, and where architectural complexity $C_a(S)$ is the sum of scaled inputs and outputs, plus the number of variables. In contrast with its predecessor model *Cognitive Functional Size* (CFS), the cognitive complexity $C_C(S)$ de-emphasizes large numbers of inputs and outputs based on weighted cognitive informatics measurements, recalculates that value as an architectural component, and uses re-calibrated cognitive weights for the calculation of the operational component.

For the CcS model of complexity, the following table is used to compute the *Basic Control Structures* (BCS). The cognitive weight of an algorithm is the degree of difficulty or relative time and effort required for comprehending a given algorithm modeled by a number of BCS. The relative cognitive weight of the *sequential* structure is assumed to be the baseline of one, that is, $w_1 = 1$.

Table 1. Calibrated cognitive weights of Basic Control Structures (BCS), reproduced from Wang (2009).

BCS	**RTPA Notation**	**Description**	**Calibrated cognitive weight (w_i)**
1	→	Sequence	1
2	|	Branch	3
3	|…|	Switch	4
4	R^i	For-loop	7
5	R^*	Repeat-loop	7
6	R^*	While-loop	8
7	↣	Function call	7
8	↻	Recursion	11
9	|| or ∯	Parallel	15
10	↯	Interrupt	22

APPENDIX B – SCATTERPLOTS FOR THE OBJECTIVE MEASURES OF COMPLEXITY

This section presents the exhaustive list of the scatterplots and regression lines for the seven objective models of complexity, as they predict the performance and anticipation scores.

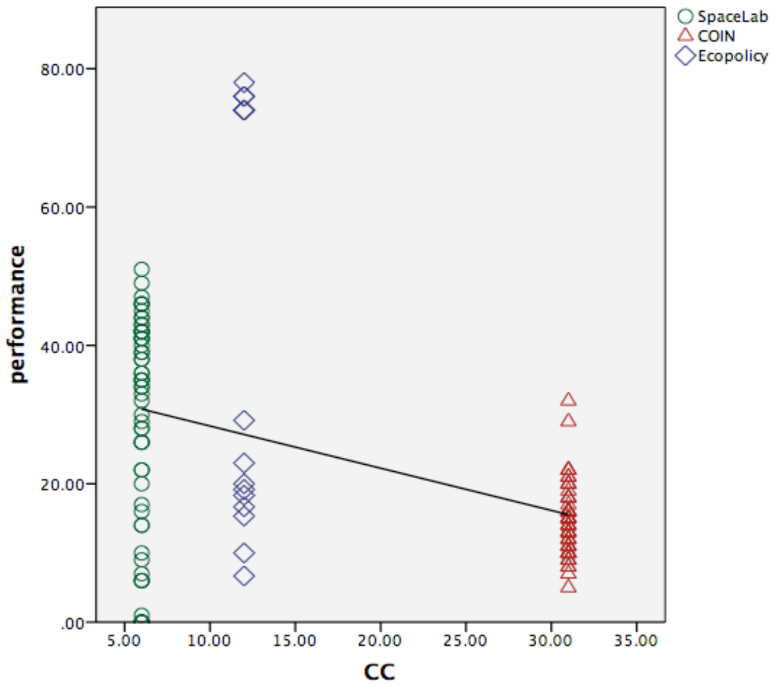

Figure 1: Scenario performances according to the Cyclomatic Complexity measure.

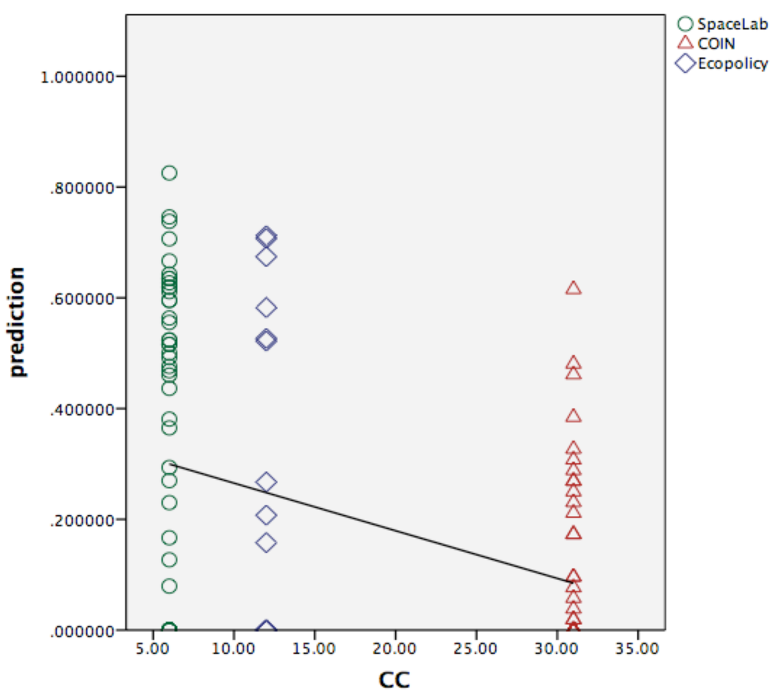

Figure 2: Prediction scores according to the Cyclomatic Complexity measure.

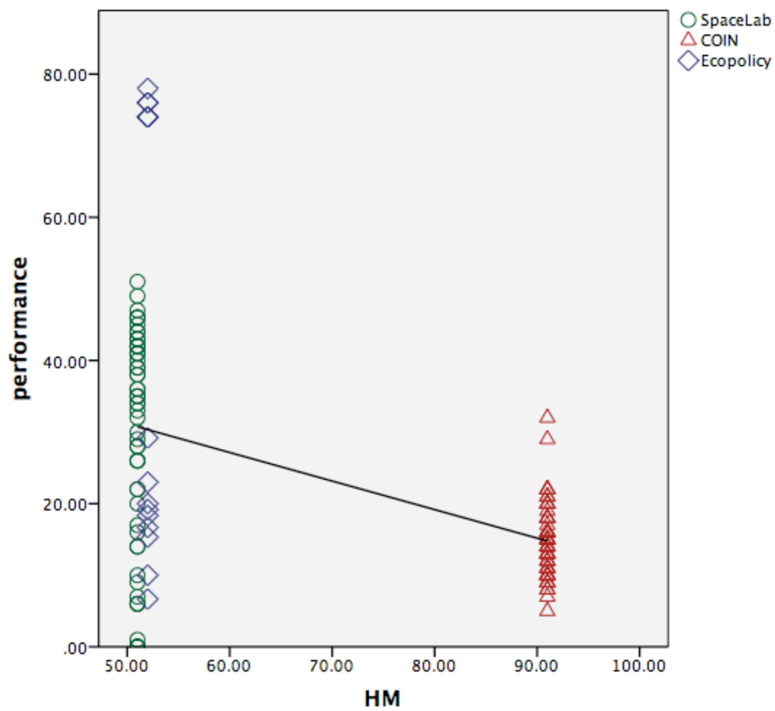

Figure 3: Scenario performances according to the Halstead Metrics measure.

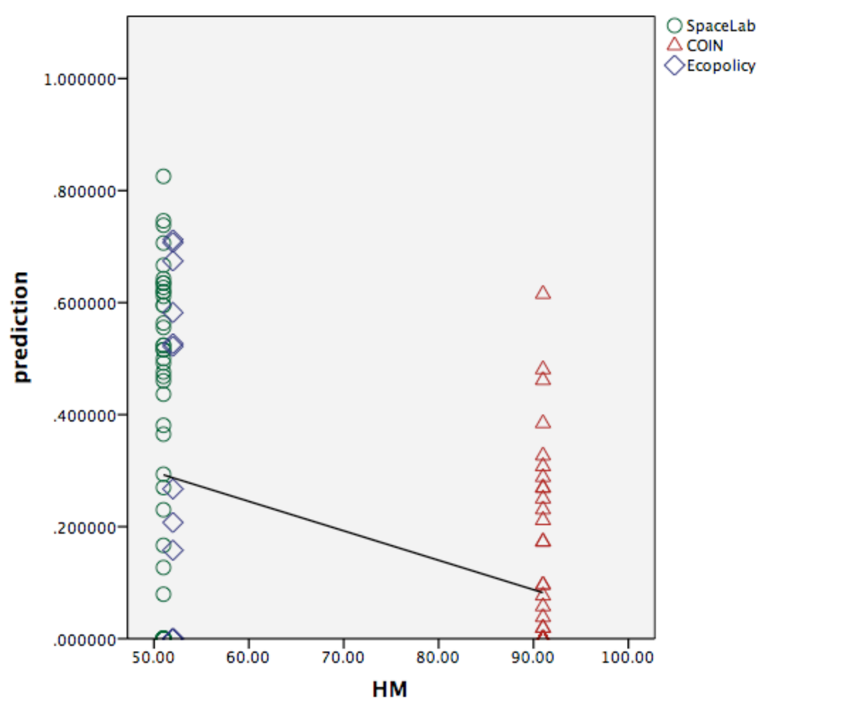

Figure 4: Prediction scores according to the Halstead Metrics measure.

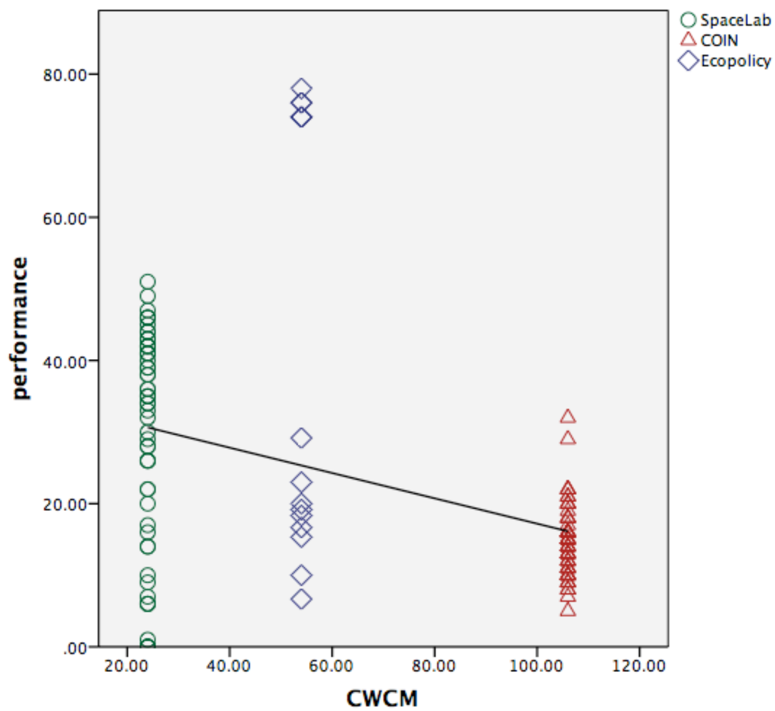

Figure 5: Scenario performances according to the Cognitive Weight Complexity Measure.

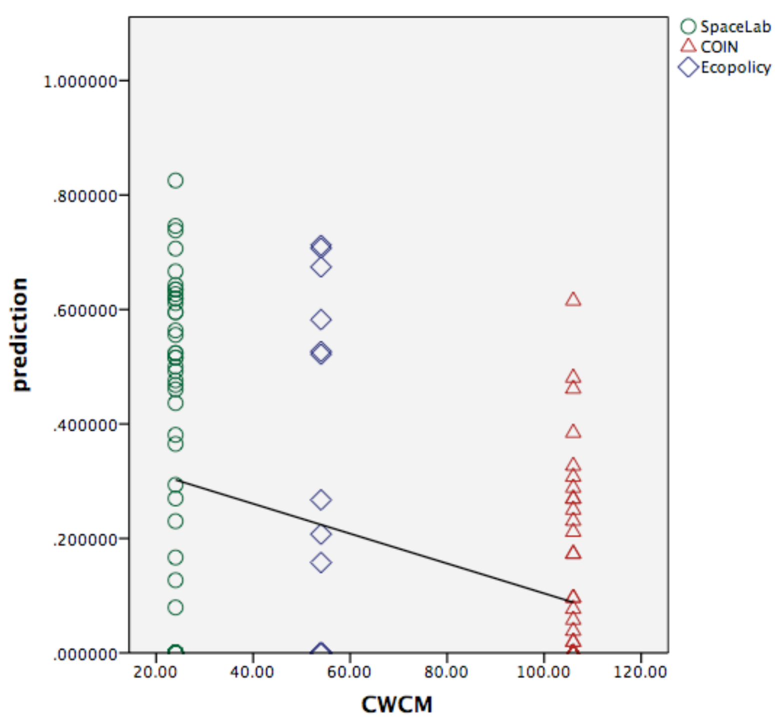

Figure 6: Prediction scores according to the Cognitive Weight Complexity Measure.

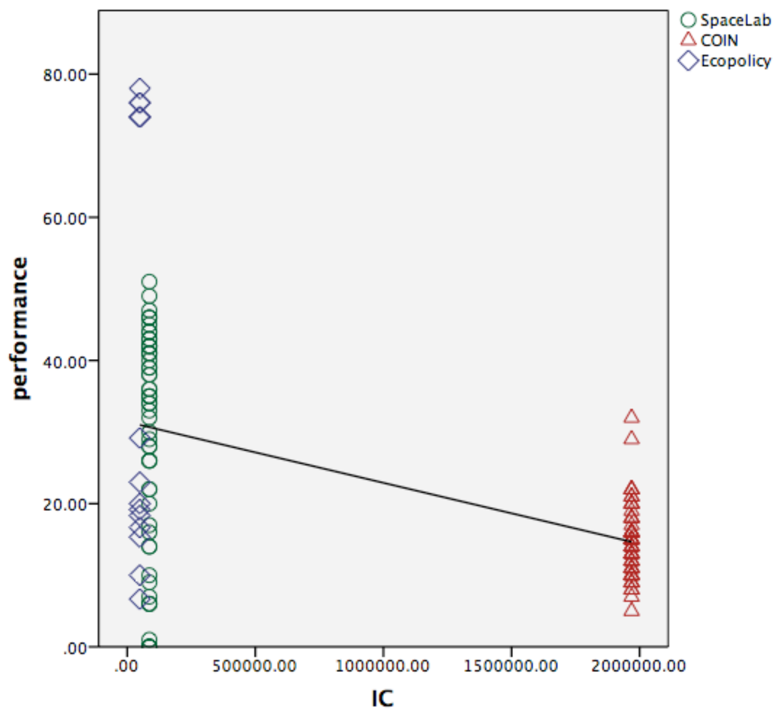

Figure 7: Scenario performances according to the Interface Complexity measure.

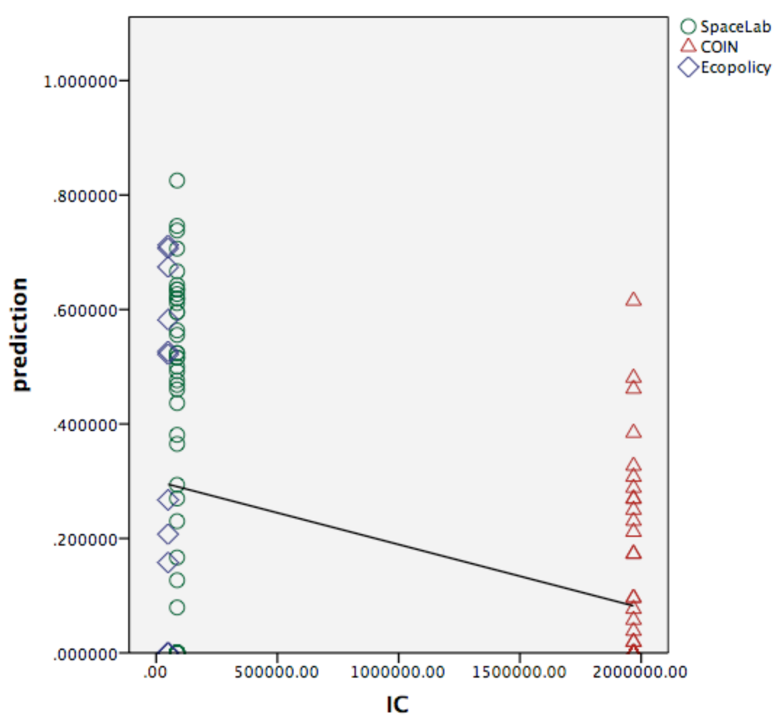

Figure 8: Prediction scores according to the Interface Complexity measure.

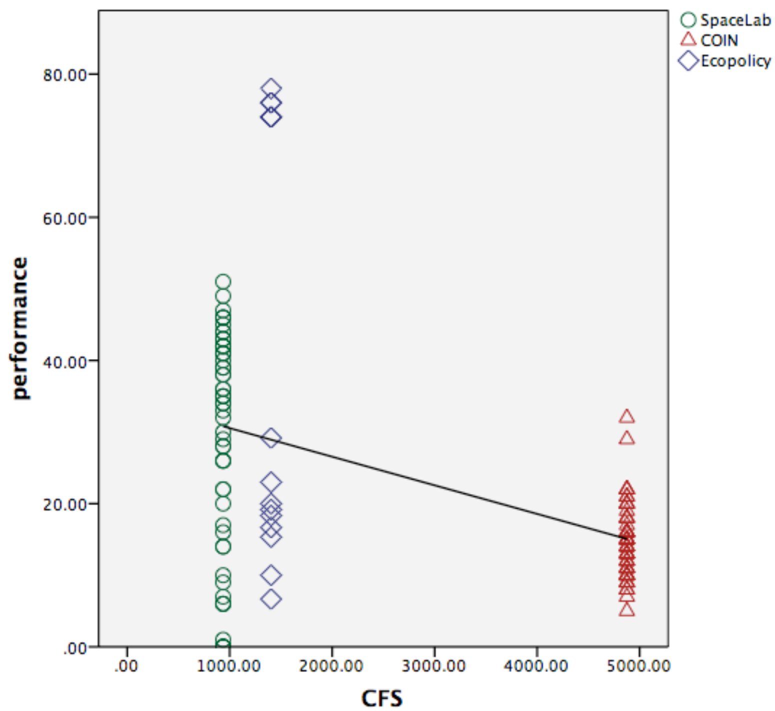

Figure 9: Scenario performances according to the Cognitive Functional Size measure.

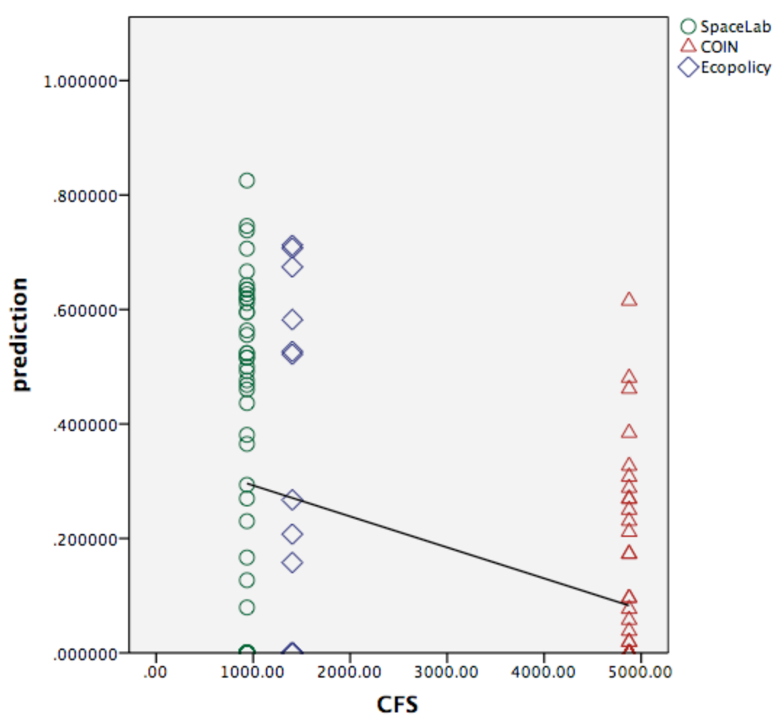

Figure 10: Prediction scores according to the Cognitive Functional Size measure.

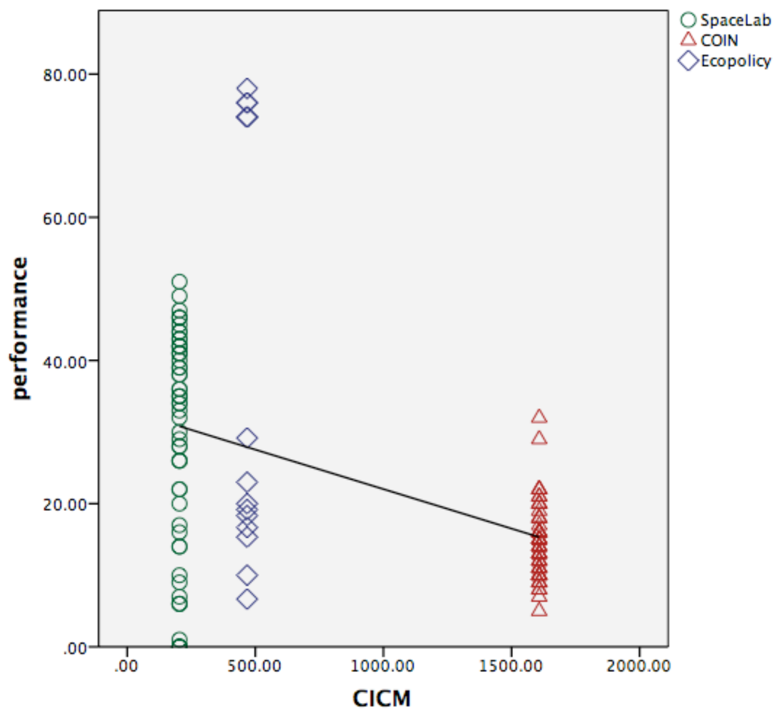

Figure 11: Scenario performances according to the Cognitive Information Complexity Measure.

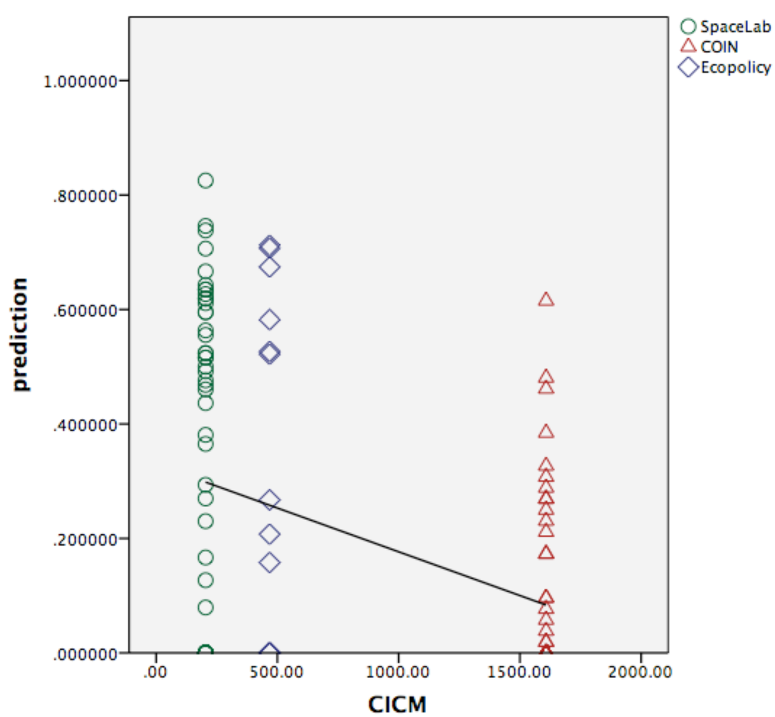

Figure 12: Prediction scores according to the Cognitive Information Complexity Measure.

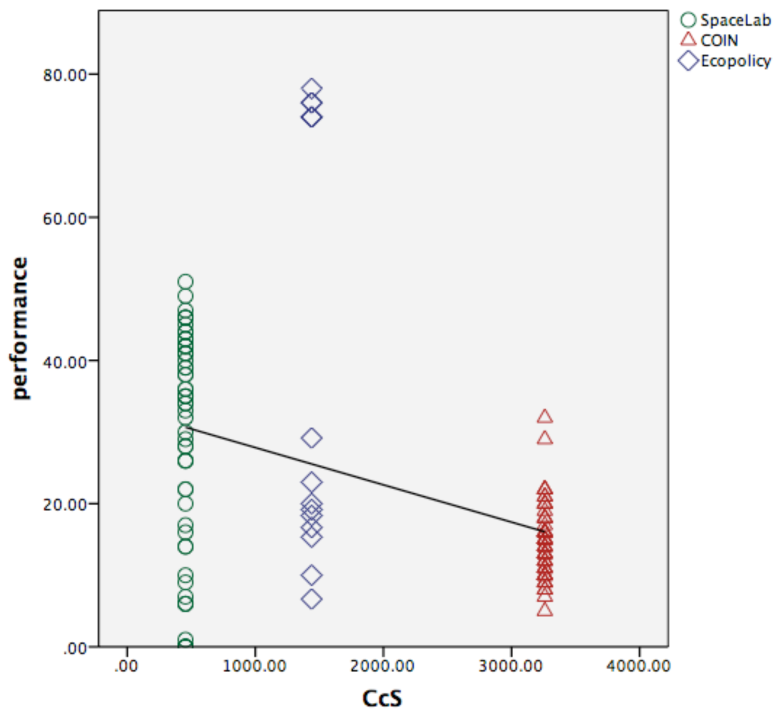

Figure 13: Scenario performances according to the Cognitive Complexity measure.

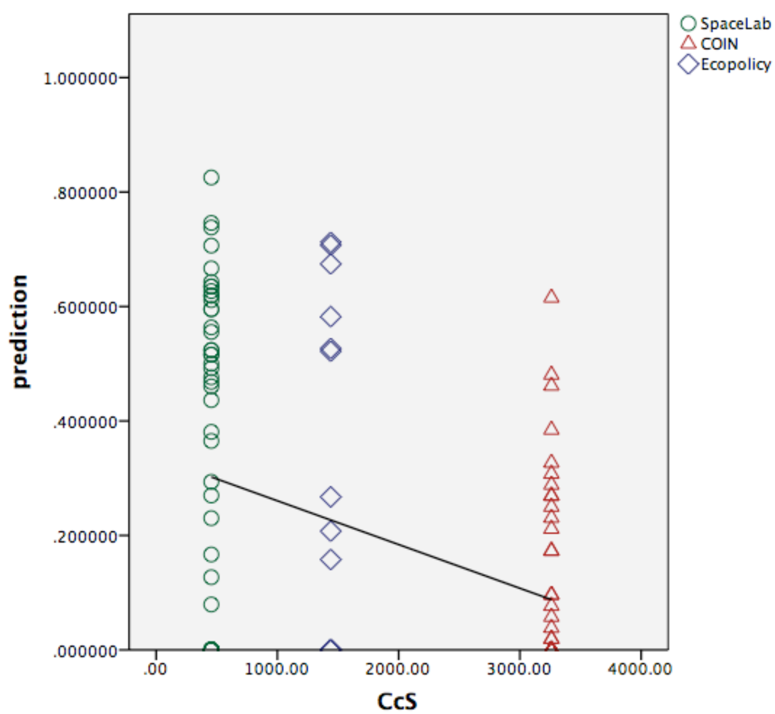

Figure 14: Prediction scores according to the Cognitive Complexity measure.