

### Codebook: Complex Relationship Variables

This codebook describes the coding rules for content analyzing cognitive maps of systems for relationships between entities that are typical of complex systems. Specifically, in this study this codebook is to be applied to cognitive maps of the analog game, *Dominion*.

Coding involves identifying whether data match conceptualizations of complex types of relationships. Data in this dataset has several parts: a) descriptions of entities, b) arrows drawn between entities to indicate relationships, and c) descriptions associated with these arrows to describe these relationships in more detail.

First code the relationships between two descriptions of entities. These are *tie-level* variables. To code this data, first read and fully understand the two descriptions of entities that are being connected. Then, consider the direction of the arrow connecting these entities. Read and fully understand the description of the relationship between the entities, taking into account this arrow. The written description should take primacy over the direction of the arrow. Next, consider how the two descriptions of entities are related in light of the arrow and relationship description. Finally, considering the three tie-level variables, code 1 or 0 for presence or absence of these three variables. Because each relationship may describe multiple of these relationships, multiple tie-level variables may be coded 1.

After coding all tie-level variables, next code relationships that potentially include more than two descriptions of entities. These are *structure-level* variables. Structure-level variables identify ties, described above, that are part of larger cognitive map structures. Examine the photo of each cognitive map in turn. Use transcriptions as the final source for interpreting the writing in these maps. Examine each tie within the cognitive map in turn to decide whether it may be a part of one of the larger structures identified below under Structure-level Variables. Assess the first structure-level variable (Indirect) before the second (Feedback). If it might be part of one of these structures, examine the descriptions of entities, arrows, and descriptions of relationships—as described above—for *every* tie that may be a part of these structures. Considering all of these,

code 1 or 0 for presence or absence of these two variables. Because each relationship may be a part of multiple structures, multiple structure-level variables may be coded 1.

An important concept in the coding rules below involves **semantic connection**. Semantic connection is based on what is written in descriptions of the relationships between entities, and is the explicit reference to these entity descriptions within the relationship description.

### Tie-level Variables

Tie-level variables are identified between two descriptions of entities connected by one arrow. This data has been transcribed.

#### Variable 1: Unlabeled

Some relationships between descriptions of entities were indicated with an arrow, but were not described. In this case, code a 1. If a relationship is coded as a 1 for this variable, do not code it a 1 for any other variable.

When to use: Any undescribed relationship.

When not to use: If there is any description of the relationship, code 0.

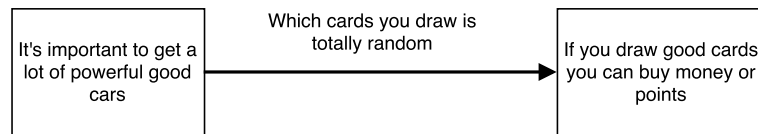
1 = Unlabeled

0 = Labeled

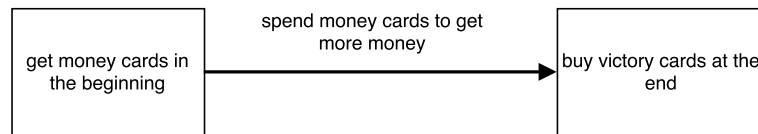
### Variable 2: Stochastic

Stochastic relationships are relationships between entities that involve randomness. Because of randomness, stochastic relationships are non-deterministic. A given event, action, or input may yield a range of possible consequences, outcomes, or outputs. Which outcome, the magnitude of the consequence, or the number of outputs may potentially have a knowable—or potentially even fixed—probability distribution. Stochastic relationships may be linear or nonlinear (see variable 4). Whereas linear/nonlinear relationships have to do with changing input-output ratios, stochastic relationships have to do with randomness.

When to use: Any mention of randomness, variable or multiple outcomes or consequences, probability, or chance in a tie description should be coded 1. If randomness, variable or multiple outcomes or consequences, probability, or chance are mentioned in an entity description, code 1 if and only if some part of this description is semantically connected in the link description. Otherwise, code 0. Example:



When not to use: Any relationship description that does *not* mention the above. Descriptions of ordered events or outcomes, such as “first” or “second,” should be coded 0. Conditional statements, such as “if a then b, but if x then y,” should be coded 0. When in doubt, code 0. Example:



1 = Stochastic relationship

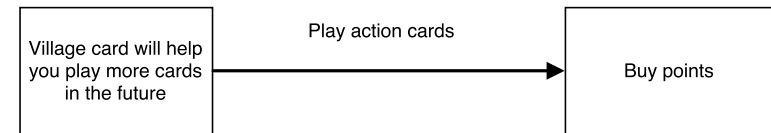
0 = Non-stochastic relationship

This codebook was used to analyze the mental model maps in Wasserman, J. A., & Banks, J. (2017). Details and dynamics: Mental models of complex systems in game-based learning. *Simulation & Gaming*. Advance online publication. doi:10.1177/1046878117715056

### Variable 3: Delayed

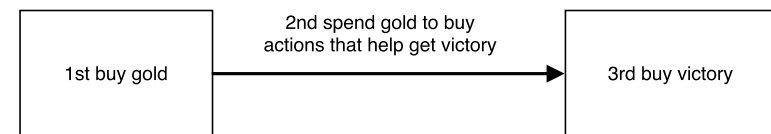
Delayed relationships are relationships that are separated temporally. In a delayed relationship, the consequence, outcome, or output of an event, action, or input occurs not instantly, but at some later point in time. This time lapse is the central characteristic of delayed relationships.

When to use: Any relationship description that mentions the intervening time after an event or input and before some outcome or output, including words that reference the passage of time like “later.” If delay is described in a node, code 1 if and only if the node is semantically connected to a tie description. Example:



When not to use: Any relationship that does *not* mention the above. Simply mentioning a temporal order (e.g., “first,” “second,” “after”) is not enough to qualify as a delayed relationship. When in doubt, code 0.

Example:



1 = Delayed relationship

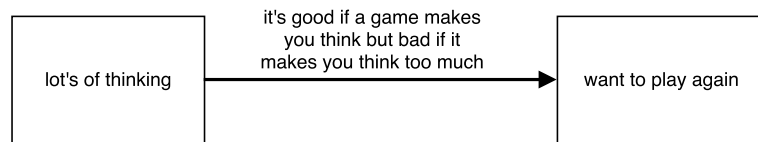
0 = Non-delayed relationship

This codebook was used to analyze the mental model maps in Wasserman, J. A., & Banks, J. (2017). Details and dynamics: Mental models of complex systems in game-based learning. *Simulation & Gaming*. Advance online publication. doi:10.1177/1046878117715056

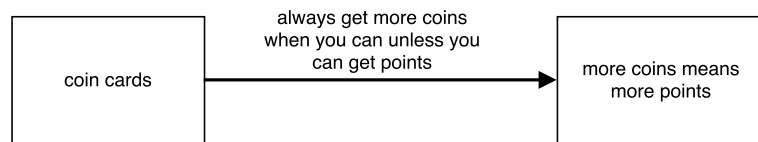
#### Variable 4: Nonlinear

Nonlinear relationships are those in which outcomes are disproportional to the input. In other words, the magnitude of an outcome or the number of outputs changes at a different rate depending on the magnitude of the event or input. Example of nonlinear relationships include exponential and logarithmic relationships. Nonlinear relationships may be stochastic or deterministic (see variable 2). Whereas stochastic relationships have to do with randomness, linear/nonlinear relationships have to do with changing input-output ratios.

When to use: Any description of dis-proportional relationships in relationship descriptions, including mentions of small inputs and large outputs or vice versa, should be coded 1. May be described in terms of proportions, comparisons of quantities/magnitudes, or words like “more and less.” Some difference between magnitude of inputs and outputs (e.g., small input, large output, or vice versa) must be stated or strongly implied. If this disproportionality is mentioned in an entity description, the relationship should be coded 1 if and only if that relationship description semantically connects to the entity description. Example:



When not to use: Any relationship that does *not* mention the above. Simply mentioning a magnitude or quantity—or even multiple magnitudes or quantities—does not qualify. When in doubt, code 0. Example:



1 = Nonlinear relationship

0 = Non-nonlinear relationship

This codebook was used to analyze the mental model maps in Wasserman, J. A., & Banks, J. (2017). Details and dynamics: Mental models of complex systems in game-based learning. *Simulation & Gaming*. Advance online publication. doi:10.1177/1046878117715056

This codebook was used to analyze the mental model maps in Wasserman, J. A., & Banks, J. (2017). Details and dynamics: Mental models of complex systems in game-based learning. *Simulation & Gaming*. Advance online publication. doi:10.1177/1046878117715056

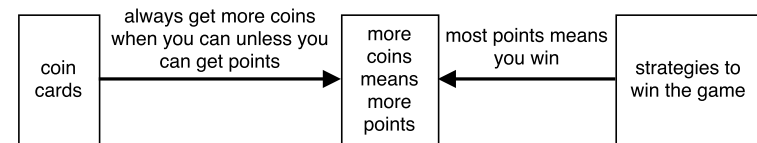
### Structure-level Variables

Tie-level variables are those that *cannot* be identified by examining solely the relationship between two descriptions of entities connected by one arrow. Instead, it requires analyzing participants' overall cognitive maps more holistically. Structure-level variables incorporate more than one relationship between entities. Although this data has been transcribed, photographs of these cognitive maps must be examined to identify these larger structures and triangulate between the overall structure and the individual entities and relationships that comprise it.

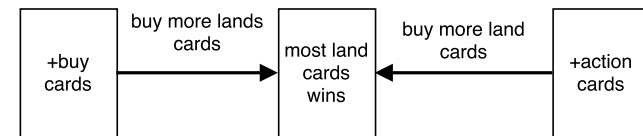
### Variable 5: Indirect

Indirect relationships are those in which an entity influences another entity via at least one other. Instead of A directly influencing B, A influences X, and X influences B.

When to use: In order for 3+ entities to be indirectly related, all entities must be semantically connected to relationship descriptions; i.e., the relationships between A → X must be semantically connected (both nodes to the tie), and the relationship between X → B must be semantically connected (both nodes to the tie). Moreover, these relationships must be directional and in the same direction, such that A influences X and X influences B. The direction of relationships described by text takes primacy over the direction of relationships implied by arrow directions. The same rules apply to indirect relationships that involve more than 3 entities, because they are all composed of 3-entity indirect relationships. Indirect relationships that begin and end with the same entity should be coded 1; for example, A → X → A. Example:



When not to use: Multiple entity descriptions merely connected by a series of arrows does not qualify—entities must be semantically connected in the relationship description. Additionally, two entities influencing the same entity does not count (A → X <- B). When in doubt, code 0. Example:



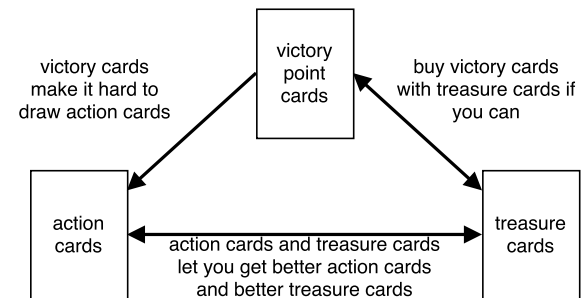
1 = Indirect relationship

0 = Non-indirect relationship

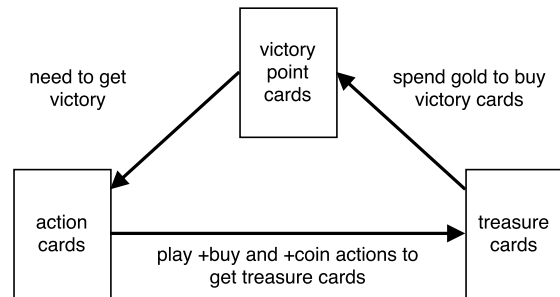
#### Variable 6: Feedback

Feedback loops involve an entity influencing itself. An entity may directly influence itself. An entity may also influence itself indirectly (see variable 5). The direction of relationships described by text takes primacy over the direction of relationships implied by arrow directions.

When to use: To count as a feedback loop, all entities must be semantically connected to relationship descriptions, and the direction of these relationships must be able to be traced *away* from an entity description and then back *to* the same entity description. The direction of relationships described by text takes primacy over the direction of relationships implied by arrow directions. Feedback loops may include only one entity that influences itself, two entities that influence each other, or 3+ entities that indirectly influence each other. Relationships may be coded 1 for both Indirect and Feedback. For potential feedback loops that include 3+ entities, the full rules in Indirect must also be followed. Example:



When not to use: A loop of entities merely connected by a series of arrows does not qualify—semantic or logical relationships linking them must be present. Relationships that are not in the same direction should be coded as 0; for example,  $A \rightarrow X \leftarrow B \rightarrow A$  should be coded 0. When in doubt, code 0. Example:



1 = Feedback loop relationship

0 = Non-feedback loop relationship

#### Other Variables

Do not code for these variables during the regular coding process. These variables will be assigned after regular coding.

#### Variable 7: Simple

Relationships between entities can be simpler than any of the complex relationships described above. Often, individuals perceive or conceive of relationships as simple—direct, linear, immediate, and deterministic—even when they are not. These simple relationships are the complete absence of the complex relationships described above.

When to use: If variables 1 – 6 are 0, code 1.

When not to use: If one or more of variables 1 – 6 are 1, code 0.