Seminar Title: **A Shape-Based Computational Model**
Presenter: Matt Boardman, B00068260

ABSTRACT

Research Approach

This seminar shows the work of Bottini et al [1] to create a so-called "Shape Completion System," a computability framework with a shape-based syntax, and to mathematically prove that it is Turing-complete. Since the pictorial representation in the model is pure, this can be considered proof that Interactive Visual Languages (IVL) [2] have computational power equivalent to a Turing Machine.

Introduction

IVL attempt to alleviate issues surrounding communication between human and machine. Some examples of familiar Shape Completion Systems are the game of Dominoes, a well-known game in which players match numbered tiles; ТЕТРИС (Tetris) [5], a popular computer game originating from Soviet-era Russia, in which a player moves and rotates a cascading series of shapes in order to minimize needed space; and Jigsaw puzzles, in which players use randomized picture fragments to assemble a photograph. Similar concepts may also employed in the fields of pattern recognition and image processing, in which algebraic characterizations of images or patterns are created [1]; and urban planning or industrial manufacturing [6], in which shapes are combined in such a way as to consume a minimum of space or material. A system quite similar to the one presented in [1] but using DNA replication as an underlying, biologically-plausible shape completion mechanism was presented in [4], which included one of the same authors, Gheorghe Păun.

Shape Completion Systems

A shape completion system, as presented in [1], is an implementation of an IVL [2] in which a list of shapes, called "polyominoes" and made from orthogonally-connected squares, are adjoined to form larger shapes. By associating a label with each shape, a "language" of words can be built, with semantic control provided by an optional "sequence mapping" that provides a transitional mapping of allowed shapes to be placed, based only on the previously placed shape. Representation consists of a quintuple:

$$\gamma = (V, P, p_0, next, lab) \quad (1)$$

where V is a finite, non-empty set of abstract symbols [1] defining an alphabet to be used to construct "words;" P is a list of one or more polyominoes from which to construct the shapes; p_0 is the first polyomino to be placed; *next* is the sequence mapping providing semantic control; and *lab* is a list of the labels associated with each polyomino in the alphabet specified by V.

We denote a particular computation to create a word as σ, the length of the resulting word w as $|w|$ and the language resulting from the set of possible words for the particular shape completion system with $L(\gamma)$. At all points in the sequence of the computation, all adjoining shapes must remain orthogonally connected, and a "correct" computation ends when the sequence of adjoining shapes forms a rectangle with no holes. A system in which $next(p) = P \;\; \forall \;\; p \in P$ (i.e. a system in which any shape can be used at any point in the sequence) is termed "simple," and a system in which *lab* contains a sparse representation in $V \cup \{\lambda\}$ is termed "reduced" [1]. Two examples of simple (but not necessarily reduced) shape completion systems from [1] are presented in the slides, with a third optional example proving RSSL–CF$\neq\varnothing$ (i.e. reduced, simple shape completion systems can generate non-context-free languages) provided as an appendix in case time permits.

Results

The authors [1] contend that, "Any finite language can be generated by a reduced, simple shape completion system" and prove this assertion mathematically. Several theorems are presented and proven in the context of the shape completion system, in order to place shape languages, simple shape languages and reduced, simple shape languages appropriately in the Chomsky hierarchy of linguistic grammar [7]. It is proven that reduced shape languages can represent any recursively enumerable grammar, and thus are equivalent in computational power to a Turing machine [1,7]. A parallel hierarchy of shape languages is presented [1], beginning with recursively enumerable languages (Type 0) as in the Chomsky hierarchy, and progressing through successive proper subsets through simple and reduced, simple shape languages to regular languages (Type 3) [7]. Several stronger variations are proposed by the authors [1] for future study, such as the specification of a *stop* polyomino to end a computation, or imposing a limit on the height of a finished rectangle. The authors [1] also propose a measure of the complexity of a calculation as the ratio of the surface area of the finished rectangle to the length of the output string.

Although shape-fitting algorithms are not discussed in the paper, several exist in current literature (see e.g. [3]). Rotation and other transformations of shapes are not mentioned, but appear to be disallowed in this representation. There is also little mention or example of the design of a sequence mapping to make such a system practical, but the authors [1] warn of potential decidability issues.

Conclusions

Complex languages can be created using shape completion systems, and such languages are Turing-complete. This paper provides mathematical proof of the "intuitive belief" [1] that pictorial representations can carry a higher information content in comparison to text-based languages.

References

[1] P. Bottoni, G. Mauri, P. Mussio, Gh. Păun, "Computing with Shapes," *Journal of Visual Languages and Computing*, 12(6), 2001, pp. 601-626.

[2] P. Bottoni, M. F. Costabile, S. Levialdi, P. Mussio, "Defining Visual Languages for Interactive Computing," *IEEE Transactions on Systems, Man, and Cybernetics – Part A*, 27(6), 1997, pp. 773-782.

[3] S. Har-Peled, Y. Wang, "Shape Fitting with Outliers," *SIAM Journal of Computing*, 33(2), 2004, pp. 269-285.

[4] L. Kari, Gh. Păun, G. Rozenberg, A. Salomaa, S. Yu, "DNA computing, sticker systems, and universality", *Acta Informatica*, 35(6), pp. 401-420, 1998.

[5] A. Pajitnov, "ТЕТРИС (Tetris): The Soviet Challenge," *Academy Soft – ELORG / Spectrum Holobyte – Sphere Inc*., 1987.

[6] M. A. Tapia, J. P. Duarte, "Shape Grammars," *MIT*, 1999 [http://shapegrammar.org].

[7] Wikipedia contributors, "Chomsky Hierarchy," "Decidability (logic)," "Free Monoid," "Recursively Enumerable Language," "Turing Completeness," "Turing Machine," *Wikipedia: The Free Encyclopedia*, 2005, [http://en.wikipedia.org/wiki].

Note: Additional references for images used in the presentation are shown in slides.