Design and Developing Methodology for 8-dot Braille Code Systems

Hernisa Kacorri[1,2] and Georgios Kouroupetroglou[1]

[1] National and Kapodistrian University of Athens,
Department of Informatics and Telecommunications,
Panepistimiopolis, Ilisia, 15784 Athens, Greece
{c.katsori,koupe}@di.uoa.gr
[2] The City University of New York (CUNY),
Doctoral Program in Computer Science, The Graduate Center,
365 Fifth Ave, New York, NY 10016 USA
hkacorri@gc.cuny.edu

Abstract. Braille code, employing six embossed dots evenly arranged in rectangular letter spaces or cells, constitutes the dominant touch reading or typing system for the blind. Limited to 63 possible dot combinations per cell, there are a number of application examples, such as mathematics and sciences, and assistive technologies, such as braille displays, in which the 6-dot cell braille is extended to 8-dot. This work proposes a language-independent methodology for the systematic development of an 8-dot braille code. Moreover, a set of design principles is introduced that focuses on: achieving an abbreviated representation of the supported symbols, retaining connectivity with the 6-dot representation, preserving similarity on the transition rules applied in other languages, removing ambiguities, and considering future extensions. The proposed methodology was successfully applied in the development of an 8-dot literary Greek braille code that covers both the modern and the ancient Greek orthography, including diphthongs, digits, and punctuation marks.

Keywords: document accessibility, braille, 8-dot braille, assistive technologies.

1 Introduction

Braille code, employing six embossed dots evenly arranged in quadrangular letter spaces or cells [2], constitutes the main system of touch reading for the blind. Moreover, assistive technologies offer a number of devices with a braille keyboard. Limited to 63 possible dot combinations per cell, according to BANA's position "there are numerous examples, both historic and modern, in which the six dots of the traditional braille cell have proven inadequate for a particular task" [4]. Increasing the number of dots from six to eight has been proposed as an extension and a number of relative standards are progressing in this direction, as shown in the related work section. In fact, BANA recognizes that 8-dot braille systems have proven to be extremely useful, particularly in the STEM fields, i.e science, technology, engineering, and mathematics. Some believe that

C. Stephanidis and M. Antona (Eds.): UAHCI/HCII 2013, Part III, LNCS 8011, pp. 331–340, 2013.

the ongoing progress in information technology will eliminate the need for embossers and braille displays that support 8-dot code. It is a fact that, although the majority of current embossers can be configured to produce 8-dot braille, the common practice is to print in 6-dot braille. Besides that, adopting an 8-dot braille code can lead to a number of challenges. However, additional advantages of rich screen displays have resulted in continuing the production of refreshable braille displays with 8-dot cells.

Despite the advancement in the domain of 8-dot braille codes for a number of languages, there is a lack of a systematic way to design these tactile systems. This work proposes a language-independent methodological approach, along with a set of principles, to assist in the development phases of new 8-dot braille codes. Furthermore, we present the results of the application of the proposed methodology for the design of an 8-dot literary Greek braille code, which covers both modern (monotonic) and ancient (polytonic) Greek writing systems, including letters, diphthongs, digits, diacritics, and punctuation marks.

2 Related Work

An 8-dot braille is only introduced as an extension to the 6-dot braille for specific tasks, where the 63 braille 6-dot cells (Fig. 1a) prove insufficient. The extended 8-dot braille character set, with 255 combinations, retains the structure of the 6-dot cell for the top 6 dots and adds dots 7 and 8, as shown in Fig. 1b. These characters are typically presented through a refreshable braille display, except for some historic examples (as described in [4]). An 8-dot braille code is not meant to substitute the 6-dot braille. This potentially raises some issues for braille readers, related particularly to the need of learning up to two codes, and switching between them. Some of these issues are: memory load, finger position adjustments, and reading style modification [17]. While the last two are related to the nature of the 8-dot braille itself, the memory load is partially dependent on the degree of similarity retained to the existing 6-dot code.

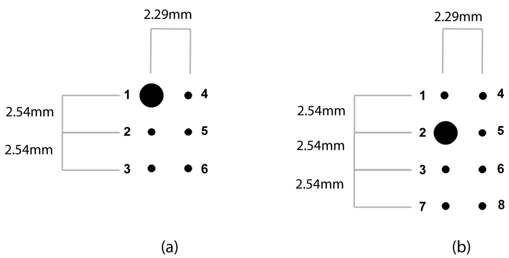


Fig. 1. Distances and labels of the dots in (a) a 6-dot braille, and (b) an 8-dot braille cell.

We can classify the existing 8-dot braille literature, as well as developed codes both for scientific and literary purposes, in two groups:

– The first group includes adopted standards related to 8-dot braille. Since September 1999, Version 3.0 of Unicode [19] has included the 256 combinations of an 8-dot braille cell (where the blank cell is counted too) with code points u+2800 to u+28ff. In addition, the encoding of 8-dot braille patterns was also supported by ISO/IEC 10646-1:2000 [8]. In 2001, ISO released ISO/TR 11548 [9-10] where it assigned semantics to the 8-dot braille patterns and produced a mapping for the Latin alphabet based character sets. In 2006, the Braille Authority of the United Kingdom added an 8-dot representation of Computer Braille in the already adopted 6-dot braille code [20].

– The second group involves other proposed 8-dot braille expansions. Some of these emphasizing on mathematical content, such as the Lambda Project [12-13], the DotPlus [18], and the GS8 [7] and others focusing on Computer Braille, e.g. [1] [3] [5] [6] [14].

3 Methodology

To successfully develop an 8-dot braille code, which could be adopted by 6-dot braille readers, we designed a methodology that results in a consistent, efficient, and unambiguous code. In this section, we define the required phases for developing the new code. In addition, we consider the design principles necessary for transitioning from a 6-dot braille code. In Section 4, we describe a successful application of our methodology to create a Greek 8-dot literary braille code.

3.1 Development Process Model

We group the development activities into three sequential phases, while incorporating feedback to resolve unexpected problems before proceeding to a subsequent phase.

First, the planning phase includes the following subtasks: (a) incorporating braille specialists and braille users of a specific language, along with its associated character set, at an early stage by recording their requirements and suggestions; (b) studying and comparing existing 8-dot braille codes with a focus on the adopted design principles and transition rules; and (c) defining the design principles for the extension of the existing 6-dot code (section 3.2).

Second, during the implementation phase, the results from the first phase are taken into account to specify the required transition rules from the 6-dot to the 8-dot braille system for the specific language. Then the following subtasks are considered necessary: (i) applying the transition rules to the existing 6-dot braille code; (ii) debugging each one rule e.g., eliminating any errors; (iii) checking for consistency between transition rules' results e.g., ensuring that application of two or more rules produces no conflicts; and (iv) providing a list of unbounded 8-dot characters available.

Finally, the evaluation phase focuses on: (1) checking the validity of the proposed code, (2) estimating its efficiency, and (3) examining its readability with users.

3.2 Design Principles

As in any system design, a crucial task in creating an 8-dot braille code for a specific language is extracting the user requirements. Braille users typically have an abstract idea of a desired result, but not how to get there. Considering users' needs, as well as other restrictions introduced by the nature of the 8-dot braille system, and the existence of a 6-dot braille code, we can recognize incomplete, ambiguous, or even contradictory requirements. We call these requirements design principles and, as mentioned in section 3.1, they have to be defined during the planning phase. Meeting these requirements is an essential milestone for the evaluation phase.

Development of the 8-dot braille code may be viewed as a problem constrained by design principles. Some principles may be contradictory, requiring a delicate balance during the specification and application of transition rules.

We propose the following design principles:

- **Compression:** map as many characters as possible to a single braille cell. Our aim is not only to save space but also to facilitate reading by avoiding the need to back-translate.
- **Intra-Similarity:** take into account the existence of a 6-dot code, which will likely coexist. It is essential that the logic behind the new code maintains ties with the 6-dot. Radical changes should only be made when unavoidable.
- **Inter-Similarity:** minimize the deviations between transition rules adopted by other (at least widespread) languages.
- **Unambiguity:** ensure that mapping of different characters to one representation only occurs when they have the same meaning or when the meaning is obvious from context. Ensure validity of the code when a combination of two or more 8-dot cells is assigned to a character.
- **Consistency:** apply the same transition rule(s) to characters of the same category e.g. capital letters. This way the mnemonic correlation between semantically related characters is taken into account.
- **Foresight:** consider possible expansions in other areas (e.g. scientific braille, computer braille) by providing unbounded cells or sharing characters.

4 Case Study – Greek 8-dot Braille Code

The National Association for the Blind in Greece estimates the number of visually impaired people in 2010 at 127.000, of which 27.000 are registered blind. A recent study [22] shows that 91,4% of the blind students in Greece use braille and 74,3% screen readers. The Literary braille code supports both the modern and ancient Greek orthography. Given the usage of diacritics, such as accents, breathings, subscripts, diaeresis, and their combinations, the majority of the symbols are mapped to two, three, or even four 6-dot braille characters. Therefore, an 8-dot extension would be beneficial for the braille display users. In this paper we propose a Greek literary 8-dot braille code following the methodology described in section 3 (see Table 1 for an overview). We arranged the presentation of the resulting code in four groups: (I) letters and diphthongs, (II) digits, (III) diacritics, and (IV) punctuation.

Table 1. Overview of the proposed Greek 8-dot braille code

Total number of characters supported	**528**
Number of characters that retained the 6-dot representation	127
Number of characters modified based on other transitions rules	387
Number of characters entirely modified	12
Number of 6-dot braille characters abolished	2
Reserved 8-dot braille characters	147

4.1 Phases of Development

We started the planning phase with a short questionnaire answered by 7 Greek professionals closely related to braille and assistive technologies for people with visual impairment. They were queried on: (i) the need for a Greek 8-dot braille code, (ii) the scope of this code, and (iii) suggestions and requirements for the design process. Valuable feedback included: minimizing the memory load on users, adopting other 8-dot braille codes supported by braille displays, and user testing before formalizing the code. We studied proposed extensions developed in other countries, discussed in the Related Work section, focusing on their common aspects such as the way they: compress capitalized letters and accents, represent digits and, map punctuation and other symbols such as the Euro symbol (€). Considering the existing 6-dot code and related 8-dot codes, we adopted the design principles proposed in Section 3.2 for the specification and application of the transition rules.

In the implementation phase, we defined five transition rules, described in section 4.2, and applied them to each of the subcategories inside the letters and diphthongs, digits, diacritics, and punctuations symbol groups. This can be seen as an iterative process: if a transition rule is applied to a subcategory, then all previously visited subcategories are checked to ensure they are conflict free. If a conflict arises then we decide which transition to preserve and modify the other subcategory accordingly. Debugging and checking for consistency were facilitated by maintaining an aggregate presentation of mapped symbols in two ways. At first, logically sorted symbols, by the subcategory and group they fit, were mapped to the 8-dot characters. Second, all the available 255 8-dot braille characters were mapped to the symbols that were represented. This last representation also allowed for easy identification of the unbounded 8-dot braille characters. The evaluation phase is an ongoing project; the results of this phase will be provided in future work.

4.2 Transition Rules

A transition rule is defined as the logic used for mapping one or more 8-dot braille characters to a symbol based on its 6-dot representation. We specified five distinct transition rules in the implementation phase. For each subcategory of symbols covered by the proposed code, one or more transition rules were applied consistently. The design principles met by each transition rule are covered below.

- **First Rule:** Retain the 6-dot representation. In this case, priority was given to the symbols mapped to one 6-dot cell. This rule was also applied in some exceptional cases where: even if a symbol was mapped to two or more 6-dot cells, the memory load of compressing it to fewer 8-dot cells would have been higher. For example, if a transition could not be applied to all symbols within a subcategory (contradicting *Consistency*). This rule focuses on maximizing the *Intra-Similarity*.
- **Second Rule:** Add dot 7 or dot 8 to the 6-dot representation. This has the effect of preserving the 6-dot representation while activating additional dots in the 8-dot cell. Restricted by the *Consistency* design principle, these rules are applied uniformly within a subcategory, e.g. capital letters. It aims to maximize *Compression*.
- **Third Rule:** Add dot 4 or dot 6 to the 6-dot representation. As in the previous rule, we preserve the 6-dot representation in the new code with the difference at the dot 4 or 6, which are activated. Similarly to the Second Rule, this satisfies the *Compression* design principle.
- **Fourth Rule:** Shift the 6-dot representation one row above or below. In this case, the 6-dot mapping is shifted vertically. This rule is required to resolve possible conflicts that arise after applying the preceding rules. Retaining a logical connection with the 6-dot representation, this supports *Intra-Similarity*.
- **Fifth Rule:** Entirely amend the 6-dot representation. Implementation of the last rule eliminates any logical connection with the 6-dot representation. It is usually required to achieve *Unambiguity* or to maximize *Inter-Similarity*.

4.3 Proposed Code

As mentioned in Section 3.2, some of the design principles contradict each other (e.g. intra-similarity and compression) and a delicate balance is required when applying the transition rules based on these principles. In this section we provide information on: (i) the symbol groups, (ii) the results after the applied transition rules, (iii) the relation to transitions adopted by other 8-dot codes, and (iv) a few examples for each of the symbol sub-categories.

Letters and Diphthongs: This group includes the small and capital letters, and the diphthongs that are common both in modern and ancient Greek. The small letters and diphthongs are the most commonly used characters and are mapped to a single 6-dot braille cell. Their 6-dot representation was retained (1^{st} rule). The capital letters, and the diphthongs with a capitalized first letter, are mapped to two 6-dot braille cells; a capital indicator and the corresponding small letters. In this case the 2^{nd} rule was applied: the indicator was removed, the 6-dot representation was copied to the 8-dot cells and, the dot 7 was raised, as shown in Fig. 2. A similar rule was applied in other proposed 8-dot extensions (e.g., [12]) and standards (e.g., [9-10]).

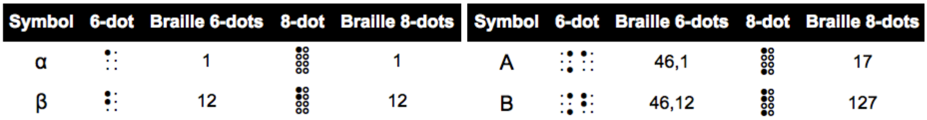


Fig. 2. Example of results of 1st rule for small (left) and 2nd rule for capital (right) letters

Digits: In the 6-dot representation of digits (0-9), a numeric indicator is used before each one of the digits. We investigated the proposed transitions in other extensions as shown in Fig. 3. However, given the restrictions of the existing Greek 6-dot braille code with previously reserved characters and the use of 8-dot for word stress, we were unable to follow any of these transitions exactly. Instead, we decided to combine part of their transition logic.

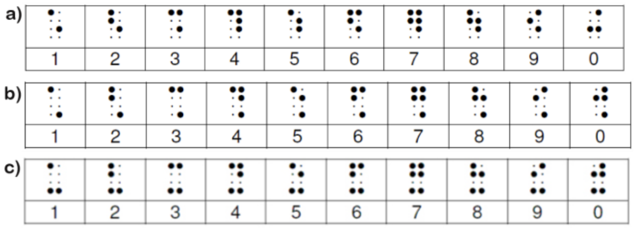


Fig. 3. Digits in 8-dot braille as adopted by: a) ISO [9-10] and most European Countries that participated in the Lambda project [12] b) Portugal [13] and c) GS braille [7].

For the 8-dot representation of digits, we first shifted their 6-dot representation one row below (4th rule) and then added dot 8 (2nd rule), as shown in Fig. 4. The result of this shifting corresponds to the digits in Nemeth braille code [15]. Given that Greece has officially adopted Nemeth for representing math [11], we minimized the memory load for braille readers by requiring them to remember only the raised dot 8.

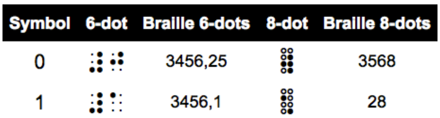


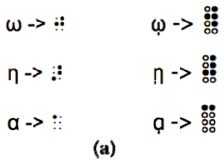
Fig. 4. Example of results of 4th and 2nd rules for digits

Diacritics: This group includes accents, breathings, subscripts, and diaeresis, including small and capital letters, and diphthongs, on which these diacritics are applied. The most commonly used accent in both monotonic and polytonic Greek orthography is the acute accent, known also as tonos or okseia. Regardless of the difference on the Unicode mapping of tonos (U+0301) from okseia (U+1FFD), these two symbols are equivalent in the Greek language. In the proposed code, the acute accent is represented by the braille dot 8. The required indicator, used before the letter where the accent applies, is removed and the dot 8 is raised in the corresponding 8-dot representation (2nd rule), as shown in Fig. 5. As for the grave accent (vareia) and the circumflex (perispomeni), their 6-dot representations were maintained (1st rule).

Symbol	6-dot	Braille 6-dots	8-dot	Braille 8-dots
´	⠐	5	⢀	8
ά	⠐⠁	5,1	⢁	18

Fig. 5. Example of results of 2nd rule for acute accent

The iota subscript (hypogegrammeni), applied solely to an alpha, eta, or omega, was represented in the 6-dot braille by an indicator immediately preceding the letter. We omitted the indicator and included the information in one braille cell, as shown in Fig. 6. For eta and omega, the dot 6 was added and for alpha, dot 4 was added (3rd rule).



(a)

Symbol	6-dot	Braille 6-dots	8-dot	Braille 8-dots
ᾀ	⠈⠁⠔	4,1,35	⠈⠉	4,14
ᾳ	⠁⠔	1,35	⠉	14

(b)

Fig. 6. Example of results of 3rd rule for iota subscript

Both rough (daseia) and smooth (psile) breathings, as well as their combinations with other accents, maintained the 6-dot representation (1st rule). The only exception is the combination of psile with okseia, which is equivalent to okseia since psile can be inferred by the context. In this case, the letters to which this combination is applied follow the okseia transition rule.

The diaeresis, appears on the letters iota and upsilon indicating that a pair of vowel letters is pronounced separately, rather than as a diphthong. However, since a diphthong is given by one braille character and a pair of vowel letters by two braille characters with each corresponding to a letter, there is no need to represent diaeresis either in 6-dot braille or in 8-dot. The only exception occurs when it is not directly applied to a letter but combined with an accent, as shown in Fig. 7. In this sense, their 6-dot representation was maintained (1st rule).

Symbol	6-dot	Braille 6-dots	8-dot	Braille 8-dots
΅	⠐	5	⢀	8
ϊ	⠊	24	⠊	24
ΐ	⠐⠊	5,24	⢊	248

Fig. 7. Example of representation of diaeresis when combined with okseia and vareia

Punctuation: Symbols in this group are punctuation marks and other symbols, such as the Euro sign (€). The 6-dot representation was maintained for most of the symbols represented by a single braille cell (1st rule). Exceptions made were a few symbols that either conflicted with the transitions results or required change to ensure unambiguity, e.g. to differentiate open with close parenthesis (in this case the 5th rule were applied). As for symbols mapped to two or more 6-dot braille characters, most of them were reassigned to one 8-dot braille cell (5th rule). There were also a few cases where the 6-dot representation was intuitive (1st rule), e.g. ellipsis. In both cases, an attempt was made to follow similar representations with other codes when a change was required.

5 Discussion and Future Work

This paper has described a language-independent design methodology for the systematic development of an 8-dot braille code. Our methodology was successfully applied in the design of a Greek literary 8-dot braille code. We found that a set of design principles, such as compression of the braille representation, similarity with the 6-dot braille code and with 8-dot codes adopted in other countries, unambiguity, and consistency, is necessary to be defined and used in the development of the transition rules from the 6-dot to the 8-dot code.

We are currently working on the validity of the proposed code and estimating its efficiency by introducing a number of evaluation metrics that will allow us to quantify the level of conformance with the design principles, e.g. metrics based on Shanon theory [16]. In future work, we intend to conduct a user study to investigate the readability of the proposed code. Whereas many factors may contribute to readability [21], we plan to investigate how users would perform when reading 8-dot characters which: (i) retain the 6-dot mapping, (ii) result from other transition rules, and (iii) correspond to a new 8-dot mapping.

Acknowledgements. This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) under the Research Funding Project: "THALIS-University of Macedonia- KAIKOS: Audio and Tactile Access to Knowledge for Individuals with Visual Impairments", MIS 380442.

References

1. American Printing House for the Blind: 8-dot Computer Braille Table (2013), `http://tech.aph.org`
2. Braille Authority of North America (BANA): Definition of Braille. English Braille American Edition 1994 (2002) (Revised 2002)
3. Cranmer, T.V.: Code for Computer Braille Notation. In: Dixon's, J. (ed.) Braille into the Next Millennium. Library of Congress, Washington, D.C (2000)
4. Dixon, J.: Eight-dot Braille, A Position Statement of the Braille Authority of North America. BANA position (2007) (Adopted September 2007)
5. Durre, K.P., Tuttle, D.W., Durre, I.: A Universal Computer Braille Code For Literary And Scientific Texts. In: International Technology Conference (1991)
6. EuroBraille 8-dot Braille Table(s) for Refreshable Braille Displays (2013), `http://www.eurobraille.com`

7. Gardner, J., Salinas, N.: The Science Access Project. GS Braille (2013), http://dots.physics.orst.edu/gs.html
8. ISO/IEC 10646-1:2000. Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane (2000)
9. ISO/TR 11548-1:2001, Ed. 1: Communication aids for blind persons–Identifiers, names and assignation to coded character sets for 8-dot Braille characters – Part 1: General guidelines for Braille identifiers and shift marks. International Standards Organization. Zurich, Switzerland (2001)
10. ISO/TR 11548-2:2001, Ed. 1: Communication aids for blind persons–Identifiers, names and assignation to coded character sets for 8-dot Braille characters – Part 2: Latin alphabet based character sets. International Standards Organization. Zurich, Switzerland (2001)
11. Kouroupetroglou, G., Florias, E.: Greek Braille Scientific Notation – Application in Information Systems for the Blind. Education and Rehabilitation Center for the Blind, Athens (2003) (in Greek) ISBN 960-87918-0-4
12. Edwards, A., McCarthey, H., Fogarolo, F.: Lambda: A multimodal approach to making mathematics accessible to blind students. In: Proc. of ASSETTS 2006, pp. 48–54 (2006)
13. Lambda Project. Linear Access to Mathematics for Braille. Device and Audio-synthesis. IST-2001-37139. Deliverable 3.7. 8-dot Maths code. Integration Table (2005)
14. Nair, N., Pitre, N., Doyon, S., Mielke, D.: Access to the Console Screen for Blind Persons using Refreshable Braille Displays. North American Braille Computer Code (NABCC) Table. BRLTTY Reference Manual (2012)
15. Nemeth, A.: The Nemeth Braille Code for Mathematics and Science Notation 1972 Revision. American Printing House for the Blind (1972)
16. Shannon, C.E.: A Mathematical Theory of Communication. The Bell System Technical Journal 27, 379–423 (1948)
17. The Science Access Project. GS Braille. Structure of Braille (2005), http://dots.physics.orst.edu/gs_braillestructure.html
18. The Science Access Project. DotsPlus Braille. Oregon State University (2005), http://dots.physics.orst.edu/dotsplus.html
19. The Unicode Standard 3.0. Braille Patterns, Range 2800-28FF (1999)
20. UK Association for Accessible Formats. Braille Computer Notation. Braille Authority of the United Kingdom 1996, Registered Charity No. 1001157, Printed by Royal National Institute for the Blind, Peterborough. Registered Charity No. 226227 (2006)
21. Wetzel, R., Amos, W.: Under the auspices of the Braille Authority of North America (BANA): Research on Readability of the Nemeth Uniform Braille System, NUBS (2009)
22. Papadopoulos, K., Koutsoklenis, A.: Reading Media Used by Higher-Education Students and Graduates with Visual Impairments in Greece. Journal of Visual Impairment & Blindness 103(11), 772–777 (2009)