Geometric Definition of the Hidden Part of a Line Drawing in a Sketch-to-Solid Methodology

Sofia Kyratzi\textsuperscript{1} and Philip Azariadis\textsuperscript{2}

\textsuperscript{1}University of the Aegean, skiratzi@aegean.gr
\textsuperscript{2}University of the Aegean, azar@aegean.gr

\textbf{ABSTRACT}

The method presented in this paper is part of a 3-step reconstruction methodology for the extraction of a polyhedron from a single view natural sketch. In particular, the current paper focuses on the 2D geometric definition of the hidden part in a topologically reconstructed line drawing (intermediate wireframe sketch) as this is generated from the given natural sketch. The proposed method is based on the different topologic relations that exist in the hidden part of the intermediate sketch. Contrarily to other approaches it employs the cross-section criterion to ensure that the final fully-determined wireframe sketch is realizable. The proposed method is successfully applied to indicative line drawings with various combinations of visible and hidden elements.

\textbf{Keywords:} computer-aided sketching, sketch-to-solid, wireframe sketch.

\section{INTRODUCTION}

A sketch is considered as a primary tool for the communication of ideas especially in the conceptual phase of design. The growing evolution and the consistent efforts to properly found the new arising Computer Aided Sketching (CAS) systems \cite{5} establish the importance of 2D sketches in design oriented disciplines. The ability of humans to realize the 3D shape of a sketched object, even if the latter appears ambiguous or contains geometric errors motivate the researchers of multiple fields, such as Computer Aided Design and Sketching, Geometric Modeling, and Artificial Intelligence, to search for efficient methods for the reconstruction of a 3D model from a given single sketch.

The subject of this paper is related to the automatic construction of a “Polyhedron from a Single Natural Sketch”. In natural sketches, the topologic and geometric determination of their hidden part underlie the main focus of the methods proposed in \cite{2}, \cite{4}, \cite{6}, \cite{8–10}, \cite{13}. In most works, a two-step reconstruction methodology is followed, where, firstly, the topologic reconstruction of a wireframe sketch is achieved on the basis of the 2D sketch information, and, secondly, a 3D polyhedron is determined from this topologically reconstructed sketch. In particular, the 3D geometry of the polyhedron is derived by exploiting the sketch topology according to an optimization procedure with soft constrains \cite{2}, or through the generation of a partial 3D object \cite{6}, \cite{13}. In most studies, the final 3D model is the result of an optimization-based procedure that weights the plausibility of each image regularity between the sketch and the inferred 3D model \cite{3}, \cite{11}. The 2D geometry of the hidden part of a wireframe sketch is not considered as a distinct problem and is only indirectly specified through the reconstruction of the 3D model.

On the contrary, our “Sketch-to-Polyhedron” methodology for the reconstruction of a 3D model from a natural sketch proceeds in three distinct steps \cite{9}. In the first step, a topological valid \textit{intermediate wireframe sketch} is produced from the given natural sketch \cite{8}. In the intermediate wireframe sketch the geometry and topology of its visible junctions, lines, and regions are precisely defined, while for its hidden elements only the topological relations are known. In the second step, the 2D geometry of the hidden elements in the intermediate wireframe sketch is specified. The second step results in a topologically and geometrically defined wireframe sketch, whose visible part is identical to the input natural sketch. In the third step, a polyhedron is created on the basis of the output wireframe sketch of the second step.

This paper focuses on the second step of the aforementioned 3D reconstruction methodology and presents a method for the definition of the 2D...
A sketch is a set of straight lines on a plane that intersect at junctions. Non self-intersecting loops of lines and junctions form the regions of a sketch. Lines, junctions and regions are called “elements” of a sketch. The sketch is considered to be the orthographic projection, on plane $\Phi: Z = 0$, of a trihedral solid (i.e., each vertex of it belongs to exactly three faces) with planar faces. The solid is considered to be in “general position” with respect to the projection plane, i.e., no face or edge of the solid is perpendicular/parallel to $\Phi$ and the adjacent faces (edges) lie on distinct planes. A “one-to-one correspondence” exists between the lines ($L$), junctions ($J$) and regions ($R$) of a sketch and the edges ($E$), vertices ($V$) and faces ($F$) of a polyhedron.

A natural sketch (1(a)) is a sketch without hidden elements and is considered to be drawn in the most informative view, i.e., there is nothing at the “back of the sketch” that cannot be directly inferred from its visible part [13], [8]. A wireframe sketch (Fig. 1(b)) includes both visible and hidden elements. On the basis of trihedral polyhedral properties, a wireframe sketch is defined as follows [7]:

**Definition 1.** A wireframe sketch is a connected graph that:

1. Every junction is adjacent to three lines (i.e., the degree of each junction $j$ is $d(j) = 3$).
2. Every line is adjacent to two regions.
3. Two adjacent regions of the sketch share exactly one line or two-or-more collinear lines.

In a natural sketch, the visible lines and junctions are categorized as internal or boundary. Internal lines ($\ell_{\text{int}}$) and junctions ($j_{\text{int}}$) follow the properties of Definition 1 (Fig. 2(a)). Boundary lines ($\ell_b$) and junctions are visible elements that are directly associated with the hidden part of a wireframe sketch (Fig. 2(a)). Boundary junctions are classified according to the number of their adjacent visible lines, in complete junctions $j_{3c}$ with degree 3, in L-junctions $j_L$ with degree 2, and in T-junctions $j_T$ with degree 1. In particular, T-junctions are visible junctions whose actual position lies on the hidden part of the sketch (Fig. 2(b)) and thus, they are considered to be adjacent to only one visible line. A boundary line that is adjacent to two L-junctions is called an “L-chain” line $\ell_L$ (Fig. 2(c)).
hidden regions (\(r_b\)) adjacent to at least one boundary line and thus to one visible region (\(r_v\)) (Fig. 2(b)). A hidden region that is adjacent to an L-chain boundary line \(\ell_L\) is called an L-chain hidden region \(r_L\) (Fig. 2(d)) and is the only hidden region \(r_L\) that is adjacent to exactly one visible line. T-junctions of a natural sketch are considered as hidden junctions in the wireframe sketch (Fig. 2(b)). Consequently, the hidden lines that are adjacent to a T-junction and belong in one visible and one hidden region are called as partially hidden lines (\(\ell_{ph}\)). An “intermediate wireframe sketch” is a minimal wireframe sketch, where (a) its visible elements are topologically and geometrically defined, and (b) its hidden part is only topologically determined.

The topologic relations in a wireframe sketch produced from a natural sketch are presented in Tab. 1. For example, the \(j_1\) junction in (Fig. 2(b)) is an L-junction in the corresponding natural sketch (Fig. 2(a)). Junction \(j_1\) is adjacent to the internal line \(\ell_1\), to the boundary line \(\ell_2\) and to the partially hidden line \(\ell_3\). Moreover it is adjacent to two visible regions \(r_1\) and \(r_2\), and to one hidden region \(r_3\). With respect to Tab. 1, \(j_1\) corresponds to the first case of the \(J_L\) row, where an internal line (\(\ell_{int}\) column) can be formed by a combination of \(\ell_{int}\) of \(j_{int}\), \(j_{int}\) of \(\ell_{ph}\) or \(\ell_{int}\) of \(\ell_{ph}\) junctions. Indeed, line \(\ell_1\) is an internal line that belongs to the case of \(\ell_{int}\) of \(j_{int}\) (resp. \(j_{int}\) of \(j_L\)). On the same manner, the boundary line \(\ell_2\) belongs to the case \(\ell_{ph}\) of \(j_{bc}\) (resp. \(j_{ph}\) of \(\ell_{ph}\)) of the \(\ell_{ph}\) column and the partially hidden line \(\ell_3\) in the case \(j_L\) of \(j_T\) (resp. \(j_{ph}\) of \(\ell_{ph}\)) of the \(\ell_{ph}\) column.

### 2.1. Notations

A line between two adjacent regions \(r_i\) and \(r_j\) is denoted as \(\ell_{ij}\). A line \(\ell_{ij}\) with terminal junctions \(v_p(x_p, y_p)\) and \(v_q(x_q, y_q)\) is written as \(\ell_{ij} : k_{ij}x + m_{ij}y + n_{ij}\), with \(k_{ij} = y_p - y_q\), \(m_{ij} = x_q - x_p\), and \(n_{ij} = x_p y_q - x_q y_p\). The number of visible/hidden lines, regions, and junctions in the sketch is respectively denoted as \(L_v/L_h\), \(R_v/R_h\), and \(J_v/J_h\).

### 3. THE CROSS-SECTION CRITERION

The cross-section criterion is a sketch realizability criterion, i.e., it asserts whether a sketch identifies with the projection of a valid polyhedron. Given a wire-frame sketch \(S\) (Fig. 3(a)) with \(L\) lines, \(J\) junctions and \(R\) regions, a cross-section of \(S\) is an arrangement of lines \(\{L_k : k = 1, \ldots, R\}\) that represents the regions \(\{R_k : k = 1, \ldots, R\}\) of \(S\). If every pair of cross-section lines...
$L_{fi}$ and $L_{fj}$ (Fig. 3(c)), that correspond to adjacent regions $R_i$ and $R_j$, intersect at a point $P_{ij}$ on the extension of the common line $\ell_{ij}$ of $R_i$ and $R_j$, the cross-section is called compatible with $S$ (Fig. 3(b)). A detailed description of the Cross-section criterion can be found in [1], [7], [12].

Whiteley [14] was the first to establish the realizability of a wireframe sketch. The authors of [12] rewrote Whiteley’s theorem as follows:

**Theorem 1.** (Cross-Section Criterion) A wireframe sketch is realizable if and only if it has a compatible cross-section, where the cross-section lines $L_{fi}$ and $L_{fj}$ of the adjacent regions $R_i$ and $R_j$ are not identical. □

On the basis of the cross section Theorem 1 the following two Corollaries can be derived.

**Corollary 1.** Given a polyhedron, two non-adjacent parallel faces $F_i$ and $F_j$ of it correspond to two parallel cross-section lines $L_{fi}$ and $L_{fj}$. □

Corollary 1 is a necessary condition for the parallelism of two faces, but it can also robustly indicate which faces intersect.

**Corollary 2.** Let $S$ be a wireframe sketch and $CS/CS'$ two compatible cross-sections of it. The intersection point $P_{ij}$, with $i,j \in \{1,...,R\}$ of two cross-section lines $L_{fi}$ and $L_{fj}$ in $CS$, and the intersection point $P'_{ij}$, of the corresponding cross-section lines $L'_{fi}$ and $L'_{fj}$ in $CS'$, are both on the extension of the same line $\ell_{ij}$. □

**3.1. Algebraic Model of the Cross-Section Criterion**

This section describes an algebraic model of the geometric cross-section criterion, combined with an algebraic representation of all the constraints that assert sketch’s realizability (i.e., cross-section compatibility with a sketch) [1]. This algebraic representation serves as a tool for an efficient computational implementation of the cross-section criterion, and it forms the basis upon which an algorithmic method is developed for the generation of a cross-section from a given sketch.

**Definition 2.** (Cross-Section Compatibility) Let $S$ be a wireframe sketch with $R$ regions, $L$ lines and $f$ junctions (Figure 4(b)). A compatible with $S$ cross-section is a set of lines $\{L_{fj}\}$ such that:

(A) Each cross-section line $L_{fj}$ that corresponds to region $R_i$, is written in the form: $b_jx + a_jy + c_i = 0$, with $i = 0,...,R - 1$.

(B) The cross-section lines $L_{fi}$ and $L_{fj}$ of two adjacent regions $R_i$ and $R_j$ are not identical.

(C) For each region $R_i$, its adjacent sketch lines $\ell_{ij}$ intersect the cross-section line $L_{fi}$.

(D) Each line $\ell_{ij}$ of $S$ that is adjacent to regions $R_i$ and $R_j$, and the corresponding to these regions cross-section lines $L_{fi}$ and $L_{fj}$ intersect at a point $P_{ij}$. □

Properties (B) and (C) are introduced, with the following Theorem 2, as constraints for the generation of a cross-section from a sketch [1].

**Theorem 2.** A set of lines$\{L_{fj}\}$ satisfies the properties (B), (C) of Definition 2 if and only if the following hold true:

(C1) $a_i b_j \neq a_j b_i,$

(C2) $k_{ij} a_i - m_{ij} b_i \neq 0,$

where $i,j \in \{0,...,R - 1\}$ correspond to the two regions $R_i$ and $R_j$ that are adjacent to the sketch line $\ell_{ij}: k_{ij} x + m_{ij} y + n_{ij}$. □

Property (D) is related to the compatibility of the cross-section with a sketch and is employed for the construction of the following cross-section system.

$\ell_{st} \leftrightarrow e_{st} : (c_s a_t - c_t a_s) k_{st}$

$\ell_{ps} \leftrightarrow e_{ps} : (c_p a_s - c_s a_p) k_{ps}$

$\ell_{pt} \leftrightarrow e_{pt} : (c_p a_t - c_t a_p) k_{pt}$

$\ell_{st} \leftrightarrow e_{st} : (c_s a_t - c_t a_s) k_{st}$

$\ell_{ps} \leftrightarrow e_{ps} : (c_p a_s - c_s a_p) k_{ps}$

$\ell_{pt} \leftrightarrow e_{pt} : (c_p a_t - c_t a_p) k_{pt}$

$\ell_{st} \leftrightarrow e_{st} : (c_s a_t - c_t a_s) k_{st}$

$\ell_{ps} \leftrightarrow e_{ps} : (c_p a_s - c_s a_p) k_{ps}$

$\ell_{pt} \leftrightarrow e_{pt} : (c_p a_t - c_t a_p) k_{pt}$
Each equation \(eqij\) sketches, where \(i, j \in [0, \ldots, R - 1]\). Thus, system (3.1) includes \(3 \times R\) unknowns and \(L\) equations. Combined with the constraints (C1) and (C2) of Theorem 2 the Cross-Section Problem (CSP) is defined. In [6] a Cross-Section Calculation Algorithm (CSCA) has been presented to obtain the unknown coefficients \((b_l, a_l, c_l)\) of the corresponding cross-section lines \(L_{fi}\). Each equation \(eqij\) corresponds to a line \(ij\) of the sketch, where \(i, j \in [0, \ldots, R - 1]\). In this paper, CSCA is expanded to solve CSP in terms of the unknown triplet \((b_l, a_l, c_l)\). An overview of the CSCA is presented below.

Let \(R_s, R_t\) and \(R_p\) be three adjacent regions of a sketch \(S\) (Fig. 4(a)). The coefficients of the corresponding cross-section lines \(L_{fi}\), \(L_{t}\) and \(L_{p}\) are included in the equations of system (3.1). The calculation of two unknown triplets \((b_j, a_j, c_j)\) and \((b_l, a_l, c_l)\), fixes the position of the cross-section lines \(L_{fi}\) and \(L_{t}\) and enables the calculation of \((b_p, a_p, c_p)\) as follows. According to Definition 2, the cross-section line \(L_i^j\); \(b_i x + a_i y + c_i = 0\) intersects with sketch line \(\ell_{pi}\); \(k_{pi} x + m_{pi} y + n_{pi} = 0\) at point

\[
\begin{align*}
\ell_{pi} &= eq_{pi} : (c_p a_q - c_q a_p) k_{pi} \\
&+ (c_q b_p - c_p b_q) m_{pi} + (a_p b_q - a_q b_p) n_{pi} = 0
\end{align*}
\]

Thus, for the calculation of cross-section line \(L_{fi}\), only two of the equations that are adjacent to region \(R_p\) are used, i.e., lines \(\ell_{pt}\) and \(\ell_{ps}\). The connected graph of a wireframe sketch allows for the development of an incremental procedure that calculates all unknowns of system (3.1) with respect to initial values provided for two unknown triplets as a starting point [7] (i.e., initialized triplets are chosen to correspond to adjacent regions and thus are associated with a single equation of (3.1)). For the determination of each cross-section line \(L_{fi}\), CSCA employs only two of the equations that include triplet \((b_l, a_l, c_l)\), leaving in total \(U_{eq} = \frac{L}{3} - 1 = 12\) “unused” equations of system (3.1). Indeed, the six initially determined values reduce the number of equations (to be used) to \(L - 1\) and the number of unknowns to \(3R - 6\). For the latter to be calculated, only \(2R - 4\) equations are employed. Combining Euler Formula \((R - L + J = 2)\) with relation \(L = 3/2J\) that associates junctions and lines in a graph of degree three, one can easily prove that the number of unused equations equals to \(U_{eq} = L - 1 - (2R - 4) = L/3 - 1\).

4. GEOMETRIC DEFINITION OF THE HIDDEN JUNCTIONS

This section focuses on the calculation of the coordinates of the hidden junctions \(j_h\) and \(T\)-junctions \(j_T\), located in the hidden part of the intermediate wireframe sketch. The number of hidden junctions \(j_h\) equals to the number of \(j_h\) and \(j_T\) in a sketch. The Hidden Junctions Geometry (HJG) method that is presented here is based on the cross-section criterion (Definition 2), Theorem 1 and on Corollaries 1&2. The input of the method is the intermediate wireframe sketch, i.e., the geometry and the topology of...
the visible part of the sketch and the topologic relations of its hidden part. The output of the method is a geometrically defined position for each hidden junction together with a cross-section that verifies the realizability of the produced wireframe sketch.

The HJCalc method proceeds in two steps. The first step focuses on the construction of a cross-section from the input intermediate wireframe sketch (see Section 4.1). In particular, a cross-section line is generated for each visible and hidden region of the sketch except for the L-chain hidden regions \( r_j \). For a hidden region \( r_j \) that is adjacent to at least two visible lines, the cross-section line \( L_{fh} \) can be defined on the basis of sketch’s visible geometry. On the contrary, for a hidden region \( r_l \), its corresponding cross-section line \( L_{fi} \) cannot be directly determined, since \( r_l \) is adjacent to only one boundary line \( \ell_i \). Thus, the cross-section line of each \( r_l \) is determined at the second step along with the geometric definition of each hidden junction (see Section 4.2). The input of the second step is the intermediate wireframe sketch and a compatible cross-section associated with it.

### 4.1. Hidden Junctions Geometry: Cross-Section Generation

The cross-section criterion is applied to the intermediate wireframe sketch. In complete analogy to system (3.1), a cross-section system (4.1) is generated from the sketch on the basis of the latter’s visible lines (Tab. 1). The boundary lines \( \ell_i \) are excluded from system (4.1) equations. The visible lines that are adjacent to a T-junction are included in (4.1), where the coordinates of the visible T-junction are used in the corresponding equations.

\[
\begin{align*}
\ell_{si} & \equiv e_{qst} : (c_1 a_3 - c_3 a_1) k_{st} + (c_2 b_1 - c_1 b_2) m_{st} + (a_1 b_2 - a_3 b_1) n_{st} = 0 \\
\ell_{ps} & \equiv e_{qps} : (c_1 a_3 - c_3 a_1) k_{ps} + (c_2 b_1 - c_1 b_2) m_{ps} + (a_1 b_2 - a_3 b_1) n_{ps} = 0 \\
\ell_{pt} & \equiv e_{qpt} : (c_1 a_3 - c_3 a_1) k_{pt} + (c_2 b_1 - c_1 b_2) m_{pt} + (a_1 b_2 - a_3 b_1) n_{pt} = 0 \\
\ell_{pq} & \equiv e_{qpq} : (c_1 a_3 - c_3 a_1) k_{pq} + (c_2 b_1 - c_1 b_2) m_{pq} + (a_1 b_2 - a_3 b_1) n_{pq} = 0 \\
\ell_{dm} & \equiv e_{qdm} : (c_1 a_3 - c_3 a_1) k_{dm} + (c_2 b_1 - c_1 b_2) m_{dm} + (a_1 b_2 - a_3 b_1) n_{dm} = 0 \\
\end{align*}
\]

System (4.1) includes \( L_w - L_L \) equations, where \( L_L \) is the number of “L-chain” lines. The equations of system (4.1) are a subset of the L equations that appear in system (3.1), since system (4.1) refers to the visible part of the wireframe sketch that is represented by (3.1). The unknowns of system (4.1) are the triplets \((b_1, a_i, c_i)\), with \( i \in [1, ..., R - 1] - \{ r_j \} \). The number of the unknowns is \( 3(R - R_L) \), where \( R_L \) is the number of \( r_l \) regions. In particular, excluding lines \( \ell_i \) from system (4.1), the triplets \((b_1, a_i, c_i)\) that correspond to hidden regions \( r_L \) are also excluded from the set of unknowns. Thus, system (4.1) includes only unknown triplets that appear to at least two equations, establishing the existence of a solution on the basis of CSCA. A cross-section is generated from the intermediate wireframe sketch and its compatibility with the sketch is evaluated with respect to a predefined accuracy level.

According to property (D) of Definition 2 the intersection point \( P_{ij} \) of cross-section lines \( L_{fi}^k; b_j x + a_j y + c_j = 0 \) and \( L_{fj}^i; b_j x + a_j y + c_j = 0 \), should lie on line \( \ell_{ij} : k_{ij} x + m_{ij} y + n_{ij} \) and satisfy:

\[
\left| \left( \frac{c_j a_j - c_i a_i}{a_i b_j - a_j b_i} \right) m_{ij} + n_{ij} \right| \leq \mu \quad \text{(4.2)}
\]

For perfect sketches that contain no error in their junctions positions it holds \( \mu = 0 \). However, small round-off errors can appear in junctions coordinates \((x + E, y + E)\) during computer-based sketching under finite precision or due to numerical processes that apply at a sketch’s digitization stage. In that case, an acceptable level of accuracy must be set for testing the cross-section compatibility with the sketch. For numerical precision errors \( E \leq 5 \times 10^{-6} \), we set an upper limit for \( \mu \) on the order of \( O(10^{-4}) \) (see related discussion in [6]). If the cross-section is found compatible, the geometric definition of the hidden part proceeds with the second step of the method, otherwise the sketch is considered as non-realizable and the process terminates.

### 4.2. Hidden Junctions Geometry: Hidden Junctions Position Calculation

The second step of the proposed methodology commences with the initial intermediate wireframe sketch and the cross-section produced during the first step. The output of this step consists of the geometric definition of the sketch hidden part together with a cross-section that verifies its realizability. This step is presented in three phases: Firstly, the geometric framework of the proposed method is detailed, followed by an algebraic representation of the geometric concept in the second phase. The presentation of the second step concludes with the algorithmic implementation of the theoretical scheme (HJCalc Algorithm) and its application to different test cases.

#### 4.2.1. Geometric framework

For the definition of the position of a hidden junction \( j_h \) (or \( j_f \)) the following procedure is applied. Let \( R_1, R_2, \) and \( R_3 \) be three adjacent hidden regions (Fig. 4(b))
that include a junction $j_h$ and let $\ell_{1,2}$, $\ell_{1,3}$, and $\ell_{2,3}$ be the three adjacent to $j_h$ hidden lines (Fig. 4(b-c)). For the sake of clarity, let lines $\ell_{1,2}, \ell_{2,3}$, and $\ell_{1,3}$ be adjacent respectively to an already known junction $j_1$, $j_2$, and $j_3$ (note: more complicated arrangements of lines are considered below). Given the cross-section lines $L_{f_1}, L_{f_2}$, and $L_{f_3}$ of the above regions, Corollary 1 establishes that each pair of these lines will intersect, respectively at points $P_{1,2}, P_{2,3}$, and $P_{1,3}$ (Fig. 4(c)). Properties (C) and (D) of Definition 2 indicate that these points must be on the extension of the corresponding sketch lines $\ell_{1,2}, \ell_{2,3}$, and $\ell_{1,3}$. Thus, line $\ell_{1,2}$ (resp. $\ell_{2,3}/\ell_{1,3}$) is defined by the point $P_{1,2}$ (resp. $P_{2,3}/P_{1,3}$) and the junction $j_1$ ($j_2/j_3$) (Fig. 4(b)). Hidden junction $j_h$ is the intersection point of $\ell_{1,2}, \ell_{2,3}$, and $\ell_{1,3}$. Corollary 2 ensures that the position of a hidden junction is the same for any cross-section generated by the sketch.

For the generation of a cross-section line $L_{f_i}$ of an L-chain hidden region $r_L$, we study two different approaches an automatic approach (Make Parallel Procedure) and an interactive approach (UDPosition Procedure). The automatic approach is based on Corollary 1 and on the fact that a sketch includes parallel lines that imply parallel faces of the corresponding 3D model (Theorem 2 in [8]). Specifically, given the geometry and topology of an intermediate wireframe sketch, the key idea is to force cross-section line $L_{f_i}$ to become parallel to a line $L_{f_k}$, where $L_{f_k}$ is the cross-section line of a visible region $r_k$, which along with $r_L$ may correspond to parallel faces $F_k$ and $F_L$. The region $r_k$ is determined as follows (Fig. 5(a)): Firstly, the visible region $r_{vL}$ that is adjacent to $\ell_L$ is identified. Afterwards, all visible lines are searched in order to find a line $\ell_k$ parallel to $\ell_L$, that belongs to at least one region $r_k$ different to $r_{vL}$. If such a line is detected, then $r_k$ and $L_{f_k}$ are selected for the definition of $L_{f_i}$. Otherwise, $L_{f_i}$ is determined through the alternative interactive approach.

According to the interactive approach, the user controls the slope of the cross-section line $L_{f_i}$ and different alternative solutions for the corresponding hidden part are obtained (Fig. 5(b-e)). However, in order to produce feasible solutions this procedure is constrained. In particular, the slope of $L_{f_i}$ is constraint by the elements that are associated with $r_L$ and the known cross-section lines that should intersect with $L_{f_i}$ (Fig. 5(b)). In the example of Fig. 5(b) the slope of $L_{f_i}$ is constraint by (i) the point $P_{1,2}$, which is the intersection point of $\ell_L$ and $L_{f_i}$ (that is already known), must also be on line $L_{f_i}$, and (ii) the point $P_{2,3}$, which is on the extension of the hidden line $\ell_{2,3}$. In particular, $\ell_{2,3}$ and the adjacent not-specified hidden junction $j_h$ belong in region $r_L$, thus $P_{2,3}$ must be on both $L_{f_i}$ and $L_{f_k}$. Since, the line $L_{f_k}$ is already known, the different slopes of $L_{f_i}$ are obtained by sliding the point $P_{2,3}$ along line $L_{f_k}$ (Fig. 5(b-d)). These different slopes of $L_{f_i}$ result in solutions for the hidden part associated to the junction $j_h$.

4.2.2. Algebraic representation of geometric framework

Following the solution of system (4.1), a cross-section system (4.3) is generated from the intermediate wireframe sketch on the basis of the hidden lines (Tab. 1). System (4.3) includes equations that correspond to (i) the boundary lines $\ell_L$ that were excluded from system (4.1), and (ii) each visible line that is adjacent to a T-junction. In this system, T-junctions are considered as hidden junctions and their true position in
Tab. 2: The equation types of system (4.3) are generated on the basis of the system’s unknowns. The table lines show the number of unknown pairs \((x_h, y_h)\) or \((x_T, y_T)\) or unknown triplets \((b_l, a_i, c_l)\) in each equation type and the total number of unknowns in the equation.

<table>
<thead>
<tr>
<th>Eq. Types</th>
<th>Unknowns</th>
<th>Type 0</th>
<th>Type 1</th>
<th>Type 2</th>
<th>Type 3</th>
<th>Type 4</th>
<th>Type 5</th>
<th>Type 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>((x_h, y_h)) &amp; ((x_T, y_T))</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>((b_l, a_i, c_l))</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Total Number of Unknowns</td>
<td>0</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>3</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

The wireframe sketch is to be calculated:

\[
\ell_{qn} \leftrightarrow e_{qdn} : (c_n a_q - c_q a_n)(y_{h0} - y_0) + (c_q b_n - c_n b_q) \\
\times (x_0 - x_{h0}) + (a_n b_q - a_q b_n)(x_{h0} y_0 - y_{h0} x_0) = 0
\]

\[
\ell_{qd} \leftrightarrow e_{qdn} : (c_n a_q - c_q a_n)(y_{h0} - y_1) + (c_q b_d - c_d b_q) \\
\times (x_0 - x_{h0}) + (a_n b_d - a_d b_n)(x_{h0} y_1 - y_{h0} x_1) = 0
\]

\[
\ell_{dn} \leftrightarrow e_{dqn} : (c_n a_d - c_d a_n)(y_{h0} - y_3) + (c_d b_n - c_n b_d) \\
\times (x_3 - x_{h0}) + (a_n b_d - a_d b_n)(x_{h0} y_2 - y_{h0} x_2) = 0
\]

......

\[
\ell_{ke} \leftrightarrow e_{ke} : (c_e a_k - c_k a_e)(y_{hi} - y_{hj}) \\
+ (c_k b_e - c_e b_k)(x_{hj} - x_{hi}) + (a_e b_k - a_k b_e) \\
\times (x_{hi} y_{hj} - y_{hi} x_{hj}) = 0 \tag{4.3}
\]

4.2.3. Hidden junction calculation algorithm

This section presents the Hidden Junction Calculation Algorithm (HJCALC) that is developed for the solution of system (4.3). The input of the algorithm is system (4.3), the intermediate wireframe sketch and a corresponding compatible cross-section. The outputs are the coordinates of each hidden junction that completely define the final wireframe sketch, and the values of all triplets \((b_l, a_i, c_l)\) that define the corresponding cross-section lines. HJCALC proceeds as follows:

**Hidden Junction Calculation Algorithm (HJCALC)**

**STEP 1:** Identify Equation Types; Identify Unknown Types;

**STEP 2:** Check for equations of Type \(\neq 0\). **IF** exist continue with STEP 3, **ELSE** exit with a solution.
Fig. 6: (a) - (f) The bold marked lines indicate the sketch lines that generate an equation respectively of (a) Type 1, (b) Type 2, (c) Type 3, (d) Type 4, (e) Type 5, and (f) Type 6, (g) junctions that their unknown coordinates are of Type A, and (h) unknown pairs of Type B and C correspond to junctions that are adjacent to “L-chain” hidden regions.

Fig. 7: (a) A simple case with unknowns of Type A, (b-d) the derived system (4.3) for these sketches includes unknowns of all types and the Make Parallel procedure is employed for the determination of the included L-chain region(s), (e) system (4.3) of this sketch does not include an unknown of Type A, and the user - interaction was necessary for the determination of the hidden part, and (f) the derived system includes unknowns of all types, but for the determination of the L-chain hidden region the UDPosition was employed.

STEP 3: Type_A = Find_TypeA ((x_{hi}, y_{hi})); IF Type_A = true, Find equations of Type 1 that include (x_{hi}, y_{hi}); Solve linear system with respect to (x_{hi}, y_{hi}); Return to STEP 1. IF Type_A = false, Continue to STEP 4.

STEP 4: Type_B = Find_TypeB ((x_{hi}, y_{hi})); IF Type_B = true, Find the equation of Type 2 that include (x_{hi}, y_{hi}) and (x_{Lj_i}, y_{Lj_i}).

\( \text{IF } (\text{Calculate } = \text{Make Parallel } ((b_{Lj_i}, a_{Lj_i}, c_{Lj_i})) = \text{true, Return to STEP 1, ELSE Calculate } = \text{UDPosition}((b_{Lj_i}, a_{Lj_i}, c_{Lj_i})); \text{Return to STEP 1.} \) □

HJCALC Algorithm is an iterative procedure that terminates when all unknowns in system (4.3) are determined. During the first iteration, all unknowns
of Type A are calculated. In each iteration, the types of equations and the types of unknowns change on the basis of the calculated unknown variables. In particular, when a pair \((xhi, yhi)\) of Type A is calculated the corresponding equations change from Type 1 to Type 0 or from Type 3 to Type 1. When a solution is found, the corresponding equations change from Type 2 to Type 1 or from Type 4 to Type 3 or from Type 5 to Type 0 or from Type 6 to Type 2. In turn, the associated unknown pairs \((xhi, yhi)\) change from Type B to Type A, and from Type C to Type B. Thus, each time an unknown is calculated and at least one hidden junction becomes a junction of Type A or B. The algorithm is applied to selected sketches that include all the studied unknown and equation types as well as different combinations of them. The test case results are presented in Fig. 7.

5. CONCLUSIONS

This paper presented a framework under which the geometry of the hidden part in a wireframe sketch can be determined. Contrarily to other approaches, the employment of the cross-section criterion establishes that, for a given realizable natural sketch, the produced wireframe sketch will correspond to a valid trihedral solid model whose orthographic projection is the given sketch. The proposed system (4.3) is an algebraic representation of the geometric constraints that should be satisfied in the hidden part of a minimal and realizable wireframe sketch. For natural sketches with strong ambiguities, the hidden geometry problem is under-constrained and more algebraic, geometric, or heuristic constraints must be defined with respect to system (4.3). The “Make Parallel” and “UDPosition” procedure are two examples of such constraints. The first one is a fully-automated procedure while the second one is based on the user interaction to address ambiguities in the hidden geometry. Although the final decision for the hidden geometry relies on the user selections, all possible alternatives are constrained to conclude to a valid and realizable solution. In future work, more constraints are to be studied and employed to facilitate the hidden geometry determination process and include possible richer structures.

ACKNOWLEDGMENTS

The research of Ph. Azariadis 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) - Research Funding Program: THALES (MIS 379516), Investing in knowledge society through the European Social Fund.

REFERENCES


