

参赛队员姓名: 何婉榕

中学: 清华大学附属中学

省份: 北京市

国家/地区: 中国

指导教师姓名: 周建军、张云贵

论文题目: 基于 GeoGebra 自动作图的
几何讲题工具研究

基于 GeoGebra 自动作图的几何讲题工具研究

何婉榕

摘要: 针对常见的几何题解析存在图文缺乏互动、讲解和图形难以同步定位等问题, 设计了集成 HTML5 动态几何软件的 Win32 C++语音讲题工具, 实现自由控制讲题进度、随意双击跳转讲题、图形和文字同步互动亮显及自动同步朗读。基于 JavaScript 正则式实现几何题文本的语义理解和自动作图的产生式规则集。在按行合成语音中对数学语言进行归一化处理和图形亮显的信息提取。本研究可为 GeoGebra 集成开发和基于语义理解的自动作图应用提供参考, 可用于几何的创新性教学和知识库构建领域。

关键词: 几何教学, 语义识别, 自动作图, 语音合成, GeoGebra

目录

1 引言	3
2 应用效果	4
3 系统架构	7
4 几个关键技术的实现	7
4.1 讲题交互控制	7
4.2 语音动态可视化	8
4.3 几何语言的智能识别	9
4.4 自动作图和高亮显示	13
5 结束语	15

1 引言

中学几何题通常包含题目文本、解答文本和几何图形等内容, 涉及的几何元素较多, 需要在讲解时显示相应的几何图形。传统课堂上的“板书画图老师讲解方式”作图不标准, 不利于分享和动态变化。书本和习题册中的“静态图文讲题方式”是最原始简单的方式, 但文字讲解和图形缺少互动难以同步定位, 图形的动态演变过程也难以展示。几何画板等动态几何软件是理想的动态交互式几何教学软件, 使用动态几何软件制作课件的新型讲题方式的研究较多, 但制作成本高, 讲解和图形难以同步定位和随意跳转定位。另外, 这几种几何题的讲解方式要么没有语音讲读, 要么需要老师配音, 存在发言不标准的情况, 需要兼顾效果和成本。因此, 研究可自动文本朗读、自动作图和图文同步亮显的几何讲题工具将有助于提高教学效率和增强教学效果, 也适应人工智能时代互联网教学的趋势。

具备自动作图和数形结合功能的动态几何软件可满足几何讲题工具的图形需求。其中, GeoGebra (简称 GGB) 以其功能强大 (包含几何、代数、符号计算、表格等功能)、开源和跨平台等特点成为理想选择 (对比分析见表 1)。

表 1 常见的动态几何软件的对比

	几何画板	超级画板	网络画板	GeoGebra	JSXGraph
版权国别	美国	中国	中国	奥地利/IGI	德国
功能规模	大	较大	中	大	小
使用平台	Windows	Windows	浏览器	跨平台/浏览器	浏览器
开源	闭源	闭源	闭源	开源	开源
二次开发	积件库	软件内定制	未知	Java/JavaScript	JavaScript
收费	收费	部分收费	免费	免费	免费

GGB 的 JavaScript (简称 JS) 移植版 (Math Apps Bundle) 适合二次开发, 国内已有教学网站和几何知识库的应用案例^[1-2]。较多传统教学软件是 Windows 桌面应用软件, 通过集成 WebKit 内核 (有 WKE 和 CEF 等开源实现版本) 可在现有软件中使用 HTML5 功能和避开 IE 控件的问题, 让 C++与 JS 代码的相互调用变得容易^[3]。因此可用 C++和 JS 基于 GGB 快速开发几何教学软件。

自动作图和同步亮显需要将几何题的自然语言转化为形式化语言以便实现计算机深层语义理解。语义理解的文本处理方法主要有中文分词、规则匹配、本体匹配和深度学习等^[4-5]。其结果可采用有限状态机、产生式、谓词逻辑、语义网和本体等多种方法表示和管理^[6], 其中产生式和谓词逻辑法以其易于理解和实现的优点得到广泛应用。中学几何题及其解析内容具有较规范的数学符号语言, 适合采用规则匹配、产生式和谓词逻辑的理解和表示方法^[2,4-5]。将语义理解的结

果进一步映射到 GGB 的作图指令就可实现几何语言的自动作图^[2]。

基于以上研究成果的总结, 本文将研究从几何题自然语言到 GGB 自动作图的语义理解方法, 并将 HTML5 的 GGB 软件集成到用 VC++/MFC 开发的语音朗读软件中, 实现语音与文字图形亮显同步的几何讲题工具。

本研究的创新点主要有: (1) 讲题中“文字+语音+图形”同步互动。语音合成朗读的进度与文本行亮显及图形元素亮显同步进行, 形象地展示解题过程中几何辅助线和几何标记的动态演变过程, 构建视听双通道的认知环境。(2) 可以自由控制讲题进度, 随意跳转改变讲题流程。既可以通过“上一行”“下一行”“重复”控制讲题进度, 也可以双击任意一行跳转改变讲题流程。(3) 采用了文字自动语音合成的朗读方式。这样减少了教学过程中的重复劳动, 也避免了发音不正的问题。老师可根据教学反馈多次修改解答内容(包括解题思路、解答过程和点评等)形成精品例题, 利于互联网传播。(4) 基于 JavaScript 正则式实现几何题的语义理解和几何元素智能提取, 适合常见的几何题目和解答文本。基于 GGB 实现几何题自动作图, 并集成 HTML5 的 GGB 软件到几何朗读软件。

2 应用效果

本几何讲题工具需要用户先编辑好每道几何例题的以下两个文件:

(1) 题目和解答的文字描述, 存储成文本文本(例如“例题 1.txt”)。

(2) 用 GGB 软件绘制题目和解答的图形, 在同一目录下存成同名文件(例如“例题 1.ggb”)。其中, 对解答部分的图形设置显示条件为相应的起始行号。

然后点击讲题工具的“打开几何题目”按钮, 选择几何题的文本文件, 将展示 GGB 几何图形和题目文本, 如图 1 所示, 进入自由控制讲题状态。

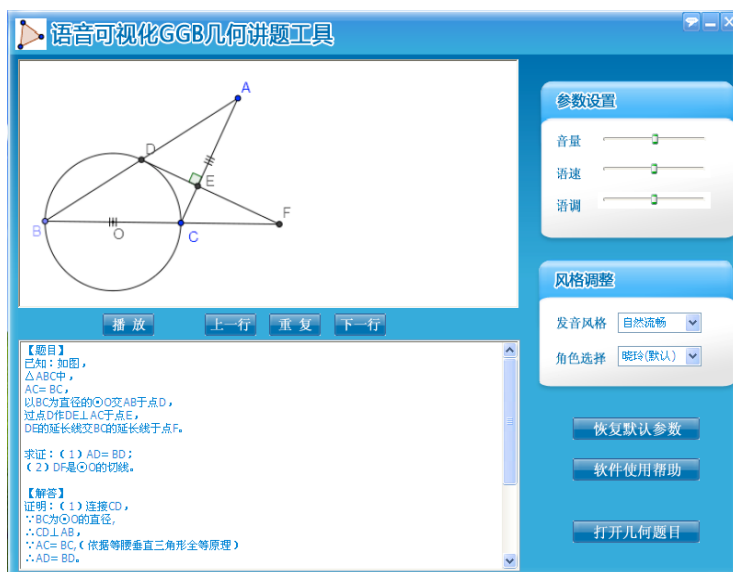


图 1 语音可视化讲题就绪界面

语音合成朗读的语音进度与图形亮显同步进行。文本区的朗读进度亮显和图形区的同步亮显效果如图 2 和图 3。图 2 中展示了题目 1 的系列亮显：三角形的亮显，垂直关系的特殊亮显，平行关系的特殊亮显。图 3 中展示了题目 2 的系列亮显：三角形相等的特殊亮显，圆的亮显。

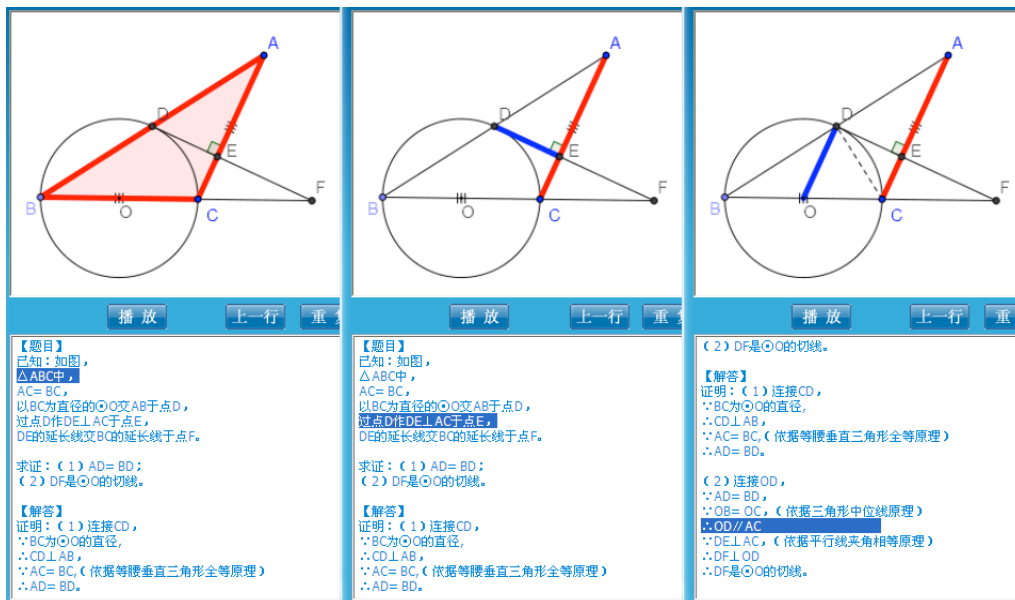


图 2 文本朗读与图形亮显同步的界面（题目 1）

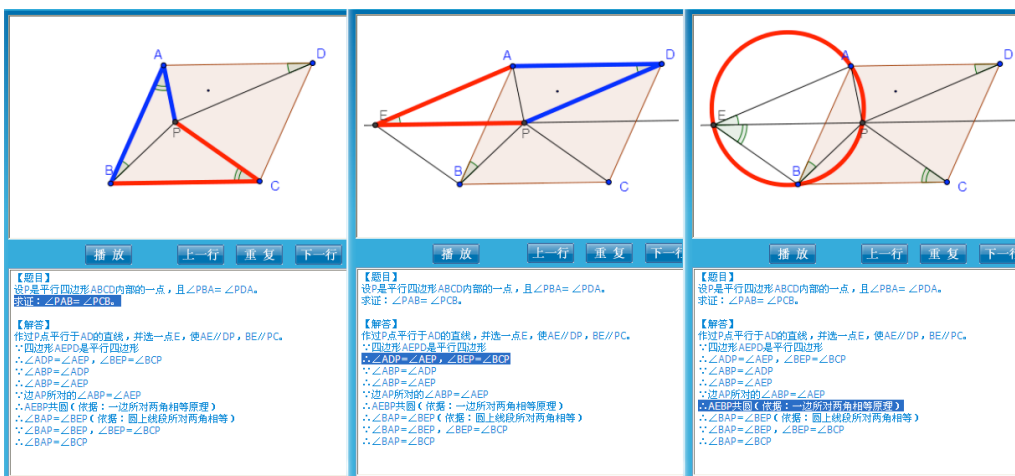


图 3 文本朗读与图形亮显同步的界面（题目 2）

本讲题方式（第 1 种）和常见的讲题方式的对比情况如表 2 所示，结果如下：

（1）第 4 种讲题方式是最原始和简单的方式，也是目前书本例题习题的主要应用方式，便于分享传播，但不够生动形象，可理解性差；

（2）第 3 种讲题方式是当前课堂教学大量使用的方式，属于一次性讲题，作图不标准，老师要改变思路就需要重新录制视频，可重复性和互联网传播性较差，老师在教学过程中进行了大量重复劳动，难以适应人工智能时代的要求；

（3）第 1 种和第 2 种基于动态几何软件的创新型讲题方式，具有“图形标

准, 讲题生动形象, 可根据教学反馈多次修改优化” 的特点, 老师可以把解题思路和解答过程进行多次修改形成精品例题。但第 1 种优势更明显, 修改题目后老师不需要重新讲题再录制视频, 具有同步语音合成朗读特点, 跳转看题更方便, 适合互联网传播, 适应人工智能时代的互联网教学的趋势。第 1 种讲题方式也会给搜题 APP 带来创造性的改变, 搜题 APP 已经积累了大量习题解答, 解答中除了常规的解题过程, 很多还有解题思路解析和考点的点评, 这些都可以成为第 1 种讲题方式的基本素材。

(4) 第 2 种和第 3 种是老师讲解的方式, 会存在少部分老师普通话不标准, 带方言口音, 学生听课费劲的情况, 第 1 种是采用语音合成的朗读方式, 很好的避免了这种情况的发生;

(5) 但第 1 种讲题方式要替代第 3 种讲题方式, 在推广上有一定难度, 需要培养大量创新型的老师, 学会制作 GGB 图形。

表 2 四种几何讲题方式的优缺点对比

讲题方式	1. 动态 GGB 语音合成同步讲题	2. 动态课件+老师讲题	3. 板书画图+老师讲题	4. 静态图形文字讲题
举例	即本文的讲题方式	老师做动态课件人工课上讲题, 搜题 APP 的动态课件讲题	传统的课堂讲题方式, 搜题 APP 的视频讲题	书本上的的习题解答, 搜题 APP 的图文解答
图形标准性	标准	标准	不标准	标准
生动形象性	非常生动形象 (完全动态同步: 语音朗读+文本区亮显+图形区亮显)	比较生动形象 (需要老师语音讲题来实现完全同步)	比较生动形象 (老师通过擦写实现几何图形的动态演变)	生动形象性差 (边看解答边找图形, 图形动态过程难显示)
重复看听性	可重复看和听, 语音合成讲题, 无需老师	可重复看, 老师语音讲题, 重听需录制视频	拍照重复看, 老师语音讲题, 重听需录制视频	无语音, 仅可重复看静态文本
跳转方便性	非常方便 (通过前一行后一行跳转, 可双击文本区某行直接跳转)	不太方便 (通过前一步后一步跳转, 但不可双击某行跳转)	不方便 (需多次拖动视频, 找到需跳转处)	方便 (简单, 看文本区跳转)
修改方便性	非常方便 (修改文件后自动生成, 无需老师讲和重录视频)	不方便 (修改完几何课件后还需老师语音重新讲题再录视频)	不方便 (需要老师改图后语音, 再录视频)	方便 (直接修改图形或文字)
动态几何	可实现	可实现	不可实现	不可实现
互联网传播	好传播 (下载本工具, 分享“GGB 图形+题目文本” 2 个小文件即可)	部分好传播 (几何课件好传播, 需录制语音部分)	不便传播 (需录像, 空间占用大, 若仅拍照等就变成了静态方式)	好传播 (简单无语音, 拍照或买书)
推广难度	需培训老师制作 GGB 图形文件	需培训老师制作几何课件, 语音讲解部分需录制视频	需录制老师讲课视频	推广容易 (已有大量静态解答)

注: 当前流行的几何动态课件主要有三种方式: PPT 课件, 几何画板课件, GGB 几何课件。搜题 APP 是指流行的几款搜题软件 (例如小猿搜题 APP、学霸君 APP、作业帮 APP), 这些软件搜到的几何题解答囊括了表 2 中的后三种讲题方式。

3 系统架构

本几何讲题工具采用如图 4 所示的 Hybrid 混合应用架构设计，在 C++ 桌面应用中内嵌 HTML5 的动态几何软件和几何语义识别模块。

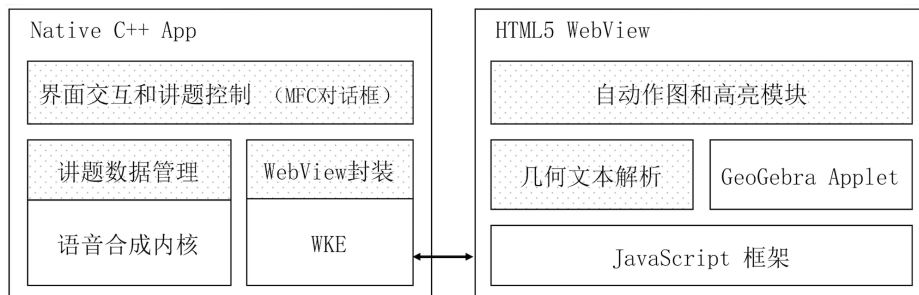


图 4 讲题工具的 Hybrid 架构

主程序基于 VC++ 2008 和 MFC 框架开发，集成了语音合成内核（TTS）模块和 WKE 浏览器内核模块。HTML5 应用基于 JS/HTML 语言和 GGB 小应用接口开发，包含了 GGB 离线运行文件（Math Apps Bundle）^[1]。HTML5 应用作为 Web 视图嵌入到对话框主界面中，由“Web View 封装”类实现窗口生命周期、窗口区域、界面显示和交互消息的传递。

通过 WKE 接口实现本地应用与 HTML5 应用的双向调用：本地应用调用 WKE 接口的“runJS(expression)”函数执行 JS 语句，绑定 C++回调函数到 HTML 的 window 对象以允许 JS 代码调用 C++函数。

对于 HTML5 浏览器内核的选型，WKE 内核（基于 WebKit 的精简版分支开源项目）体积小（动态库约 11MB）且使用简单，CEF 和 Blink 内核体积大且编译困难，经试验后本讲题工具选择了满足 GGB 集成需求的 WKE 内核。

4 几个关键技术的实现

4.1 讲题交互控制

本讲题工具除了逐行顺序朗读和暂停恢复朗读的基本控制外，还实现了跳步讲题，如**错误！未找到引用源。**所示，允许用户自由控制讲题进度。可通过点击或连续点击“上一行”和“下一行”按钮进行跳步，点击“重复”按钮重复朗读和亮显当前行，还可通过双击文本区的任意一行改变讲题流程，双击文本解答区的任何一行，即能看到图形区的同步亮显，也能听到本行的语音合成朗读。其原理是以行为基本单位进行语音合成，朗读过程中动态提取和传递关键字实现语音朗读和图形动态同步亮显。另外用户还可以不用鼠标直接用键盘快捷键控制：“空格键”执行播放暂停，“左键”执行上一行，“右键”执行下一行，“下键”执行

重复本行，更加方便用户的自由控制讲题。

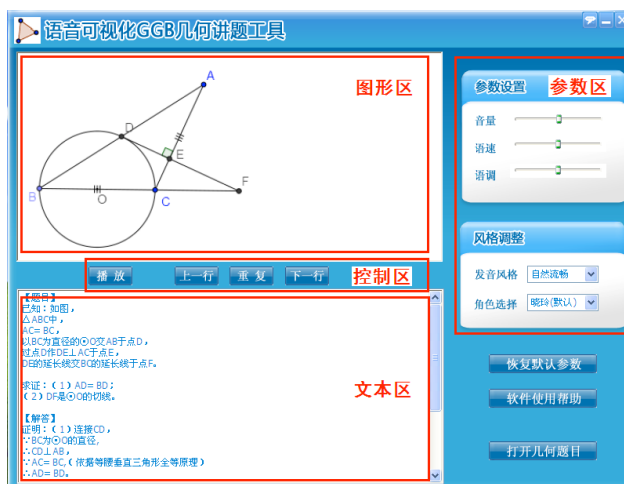


图 5 几何讲题工具的交互控制界面

在**错误！未找到引用源。**中，讲题界面有下列四个主要的交互区域。

- (1) 图形区：显示 GGB 图形，随语音朗读的进度同步亮显相应图形元素。
- (2) 文本区：即题目和解答文本区，会同步亮显当前正在合成朗读的行。
- (3) 控制区：实现了播放、暂停、上一行、下一行和重复本行等朗读进度控制，双击文本区的某行会直接朗读该行。同时提供键盘快捷控制功能：“空格键”播放或暂停，“左键”执行上一行，“右键”执行下一行，“下键”重复本行。
- (4) 参数区：可随时调整音量、语速和语调等语音合成参数。

4.2 语音动态可视化

本文的“语音动态可视化”是指几何题语音合成自动朗读时，同步亮显正在合成的几何题文本行，并同步亮显正在朗读的动态几何图形元素，构建视听双通道的认知和教学环境。本文在调用集成北京宇音天下科技有限公司的语音合成内核（emTTS 内核）的基础上，进一步研究几何题目语音合成的相关技术。

emTTS 内核是通用语音合成内核，实现了文字转语音和音量语速语调发音人等朗读参数的控制等基本的语音合成功能，主要用于新闻、文章和车载导航等的自动朗读。但对几何题目的文本朗读没有进行专门的优化，和常见的语音合成内核一样都会出现有些合成不正确的问题（例如“ $\triangle \odot \perp // \angle \therefore \therefore \cong \text{Rt}$ ”识别的正确性问题）。为了实现“语音动态可视化”，我们对 emTTS 内核进行二次开发和增强，主要解决以下三个问题：

- (1) 几何数学语言的语义理解。

对几何题目文本进行几何语义的理解，让软件能够按人的朗读方式把几何符号进行正确的语音合成朗读。几何题有图形、文字和符号等三种数学语言，我们

需进行词性识别符号识别后进行归一化处理, 让学生掌握其正确的读法是几何教学的重要内容。本讲题工具先将几何题目文本进行预处理 (例如去除空格), 然后采用基于字典和上下文匹配的符号替换进行归一化处理, 例如将 “Rt” “ ∞ ” “ $\geq$ ” “ $\odot$ ” “ $\perp$ ” “ $\therefore$ ” “ $\triangle$ ” 分别转换为 “直角” “相似于” “大于或等于” “圆” “垂直” “因为” “三角形”, 这些几何符号的语义理解的优先级高于 emTTS 内核里的语义理解。

(2) 同步亮显图形的控制。

本讲题工具在按行合成语音的过程中, 将字母序列提取为关键字 (例如对 “直线 AB 是 $\odot O$ 的切线” 将提取到关键字 “ AB ” 和 “ O ”), 将当前行号和关键字作为图形亮显的上下文定位信息 (见 4.3 节的 “四元组” 描述) 传递给 JS 的亮显模块。在 JS 的亮显模块中根据行号和关键字找到谓词, 执行相应的亮显函数 (见 4.4 节的 “动态亮显方案” 描述) 实现动态亮显。这样就实现了在恰好语音合成到这些关键字时, 同步调用亮显函数实现动态亮显 (例如对 “直线 AB 是 $\odot O$ 的切线”, 朗读到 “ AB ” 时直线 AB 会亮显; 朗读到 “ O ” 时圆 O 会亮显)。

(3) 图形关系的同步亮显特殊控制。

本工具对几何图形的中一些特殊关系进行识别, 例如 “线段 CD 与 AB 垂直” “ $AC=BC$ ” “ $\angle ADP=\angle AEP$ ” “ OD 平行于 AC ” “ $\triangle ABC \cong \triangle DEF$ ”。本文按 4.3 节所述提取 “ $\perp$ ” “ $\parallel$ ” “ $=$ ” “ $\cong$ ” (“垂直” “平行” “等于” “全等于”) 等两个几何元素之间的关系, 在亮显时采取不同的策略: 对于孤立的几何元素, 默认用蓝色进行亮显, 对于有关系的几何元素, 本文对关系中的第一个几何元素用红色亮显, 关系中的第二个元素用蓝色亮显, 且前一个元素的亮显时间一直会延迟到后一个元素亮显结束, 以便能更直观地向用户呈现几何关系。对于有较多关系角度和关系线段的几何题, 同步特殊亮显能让用户快速图文对照定位, 缩短理解时间。

4.3 几何语言的智能识别

中学几何题的自然语言是描述多样化和规律性较强的数学符号语言, 需要转换为形式化语言以便计算机能理解并实现自动作图。例如 “ $AB \perp CD$ ” 与 “线段 AB 和 CD 垂直” 是相同含义的表述。本研究采用下列三个步骤从几何题文本提取需要自动作图的几何元素。

(1) 采用易于表达的扩展巴科斯范式 (EBNF) 表述几何题模式匹配的形式化语言, 结果如图 6 所示, 包括几何符号 (例如 “ $\odot$ ” “ $\angle$ ” “ $\parallel$ ” “ $=$ ”)、特定前缀 (例如 “射线” “延长”) 和后缀 (例如 “共圆”)。针对图形亮显用途对表示图形类型的特定前缀进行裁减, 不特别识别 “ $\triangle$ ” “平行四边形” “梯形” 等多边

形前缀, 识别其顶点序列即可构建亮显的几何信息, 以此降低语法复杂性。

(2) 对 EBNF 描述进行去除非终结符的展开处理, 并采用语法图构建工具 (<http://bottlecaps.de/rr>) 进行合并排序处理, 结果如图 7 所示。此方法可避免复杂的正规文法分析工作, 复用较成熟的语法分析工具的能力。从语法图易知字母的不同组合和不同的上下文分别有点、线段、射线、三角形、四边形、多边形和延长线等语义, 从而识别多个匹配规则的冲突并采用相应的策略。

```
亮显规则 ::= 点 | 线 | 延长到 | 圆 | 角 | 多边形 | 线对 | 共线
点 ::= '点'? 大写字母
线 ::= ('射线'|'线段'|'直线')? (双字母|小写字母)|'线段'? 双字母 '的延长线'
延长到 ::= '延长' '线段'? 双字母 ('到'(点|线))?
圆 ::= ('圆'|'○') 大写字母 双字母?
角 ::= ('角'|'∠') (大写字母 双字母?| 希腊字母 | 数字)
多边形 ::= 大写字母 大写字母 大写字母+
线对 ::= 线 ((关系符号 | 关系名1) 线 | ('和'|'与') 线 关系名2)
关系符号 ::= '/' | '⊥' | '=' | '≡'
关系名1 ::= ('平行' | '垂直')'于'? | '等于' | '交'
关系名2 ::= '平行' | '相等' | '等长' | '垂直' | '相交'
共线 ::= 大写字母 (点连接符? 大写字母)+ ('共线'|'共圆')
点连接符 ::= ',' | ', ' | '、' | '和' | '与' | '及'
双字母 ::= 大写字母 大写字母
```

图 6 几何题亮显识别的 EBNF 描述

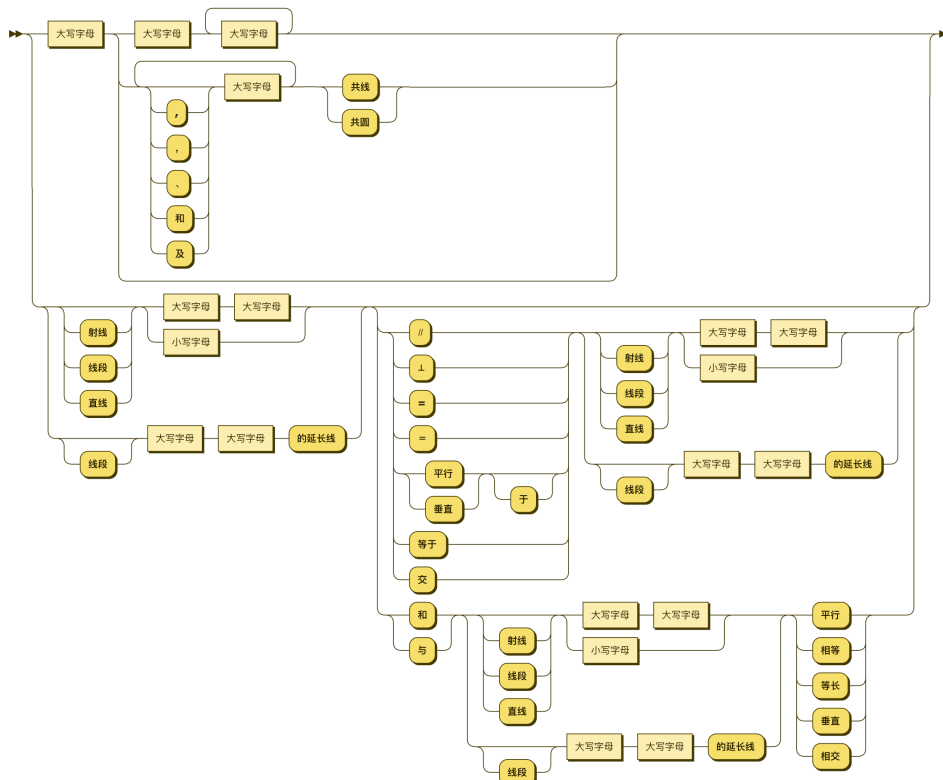


图 7 几何题亮显识别的语法图 (局部)

(3) 从已排序优化的语法图构造正则表达式, 将其作为自动作图产生式规则的条件部分。而结论部分则为自动作图的一阶谓词逻辑描述。采用规则排序的冲突解决策略, 一行文本经去除空白字符的预处理后依次应用每个规则进行规则匹配, 最先匹配成功且位置最靠前的规则为最佳匹配规则, 然后应用此规则提取谓词、关键字和关系符号, 算法见图 8 描述。表 3 列举了产生式规则的部分示例, 其中结论部分的谓词表示自动作图和亮显的类型名。对于“ $AB \parallel CD$ ”等的二元图形关系采用两个一阶谓词逻辑描述表达图形亮显信息, 不使用类似于“ $\text{perpendicular}(\text{segment}(CD), \text{segment}(EF))$ ”的二阶谓词逻辑描述图形关系是因为本几何讲读工具暂时还没有保留关系类型的必要。

表 3 自动作图的产生式规则 (局部)

条件: 正则表达式	输入文本	结论: 一阶谓词逻辑
$[A-Z]\{2\}(\text{平行}(\text{于})? //)[A-Z]\{2\}$	$AB \parallel CD$	$\text{segment}(AB, //), \text{segment}(CD)$
$[A-Z]\{2\}(\text{和} 与 、)[A-Z]\{2\}\text{平行}$	AB 和 CE 平行	$\text{segment}(AB, //), \text{segment}(CE)$
$[A-Z]\{2\}(\text{和} 与 、)[A-Z]\{2\}\text{垂直}$	AC 与 EF 垂直	$\text{segment}(AC, \perp), \text{segment}(EF)$
$[A-Z]\{2\}(= \text{等于})[A-Z]\{2\}$	$AB=CD$	$\text{segment}(AB, =), \text{segment}(CD)$
$[A-Z]\{4\}$ 共圆	四点 $ABEP$ 共圆	$\text{circle3p}(ABEP)$
$(\text{圆} \odot)[A-Z]\{3\}$	圆 ABC	$\text{circle3p}(ABC)$
$(\text{圆} \odot)[A-Z]$	$\odot O$	$\text{circle}(O)$
$(\text{角} \angle)[A-Z]\{3\}$	$\angle PAB$	$\text{angle3p}(PAB)$
$(\text{角} \angle)([A-Z][0-9][\u0370-\u03ff])$	$\angle \beta$	$\text{angle}(\beta)$
$[A-Z]\{3,8\}$	平行四边形 $ABCD$	$\text{polygon}(ABCD)$
$[A-Z]\{2\}$ 的延长线 延长 $[A-Z]\{2\}$	CD 的延长线	$\text{extension}(CD)$
射线 $([A-Z]\{2\})[a-z]$	射线 AB	$\text{ray}(AB)$
(线段)? $([A-Z]\{2\})[a-z]$	AB	$\text{segment}(AB)$
$[A-Z]$	点 P	$\text{point}(P)$

对几何题的每行文本依次进行规则匹配, 结果用四元组 (行号, 关键字, 谓词, 关系符号) 表达, 其 JSON 格式和解析示例如表 4 所示。例如第六行的“ AC 与 EF 垂直”的结果是 (6, AC , 'segment', ' $\perp$ '), (6, EF , 'segment', null)。其中“关系符号”用于延长高亮显示时间。“关键字”是匹配内容中的字母序列 (不包含完整的匹配内容是为了降低同步亮显接口的数据复杂性, 增强文字转语音引擎的可替代性), 用于同步亮显的行内上下文定位。例如原文“ $\angle PAC$ ”和“线段 AB ”的关键字分别是“ PAC ”和“ AB ”。

```
function pickKey(t) {return t.replace(/^[^A-Za-z0-9\u00370-\u003ff]/g, '');} // 提取字母和数字
function parse2Segments(text) {
    var a = text.replace(/^[^A-Z].+$/ , ''); // 提取第一个字母序列
    return [['segment', a], ['segment', pickKey(text.substr(a.length))]];
}
var rules = [ // 匹配规则集, 简单规则省略 parse 函数
    { re: /[A-Z]{2}(平行(于)?|//)[A-Z]{2}|[A-Z]{2}(和|与|、)[A-Z]{2}平行/,
      type: '//' , parse: parse2Segments },
    //...
    { type: 'circle3p', re: /[A-Z]{4}共圆|(圆|⊙)[A-Z]{3}/ },
    //...
    { type: 'extension', re: /[A-Z]{2}的延长线|延长[A-Z]{2}/ },
    { type: 'segment', re: /(线段)?[A-Z]{2}/ },
    { type: 'circle', re: /(⊙|圆)[A-Z]/ },
    { type: 'point', re: /[A-Z]/ }
];
function applyRule(queue, line, rule, text) {
    var r = rule.parse ? rule.parse(text) : [rule.type, pickKey(text)];
    if (r[0] instanceof Array) { // 如果是图形关系结果, 则分别添加两个图形的结果
        queue.push({ line: line, name: r[0][0], key: r[0][1], rel: rule.type }, { line: line, name: r[1][0], key: r[1][1] });
    } else { // 是简单结果, 则直接添加结果
        queue.push({ line: line, name: r[0], key: r[1] });
    }
}
function parseLines(lines) { // 解析几何题的文本行
    var queue = [];
    lines.forEach(function(text, line) { // 解析每一行
        text = text.replace(/\s+/g, ''); // 取一行文本, 去除空白字符
        while (text) { // 开始匹配文本, 直到没有匹配项或已到行尾
            pos = 9999; rule = null; // 匹配位置初始化为最大值
            rules.forEach(function(r) { // 顺序尝试每个规则
                t = text.search(r.re); // 记下该规则的匹配位置
                if (t >= 0 && pos > t) { // 如果匹配位置更靠前, 则为最佳匹配候选项
                    pos = t; rule = r;
                }
            });
            if (rule) { // 找到最佳匹配项
                t = text.match(rule.re)[0]; // 匹配到的文本
                applyRule(queue, line + 1, rule, t); // 提取关键字和谓词
                text = text.substr(pos + t.length); // 继续该行的后续匹配
            } else { break; } // 没有匹配项就结束本行
        }
    });
    return queue;
}
```

图 8 基于规则排序的匹配算法

表 4 几何题文本解析示例

输入文本行数组	解析结果: 四元组 (行号, 谓词, 关键字, 关系符号)
['AB // CD, AC 与 EF 垂直', '四点 ABEP 共圆', '∠PAB=β']	[{"line":1,"name":"segment","key":"AB","rel":"//"}, { "line":1,"name":"segment","key":"CD"}, { "line":1,"name":"segment","key":"AC","rel":"⊥"}, { "line":1,"name":"segment","key":"EF"}, { "line":2,"name":"circle3p","key":"ABEP"}, { "line":3,"name":"angle3p","key":"PAB"}, { "line":3,"name":"angle","key":"β" }]
['直线 EF 与 ⊙O 相切于点 M']	[{"line":1,"name":"line","key":"EF"}, { "line":1,"name":"circle","key":"O"}, { "line":1,"name":"point","key":"M" }]

4.4 自动作图和高亮显示

基于动态几何软件实现几何知识自动作图的研究集中在实现全自动智能作图^[2,5]。但受自然语言复杂性和图形位置难以预测的影响, 存在图形不够美观和系统过于复杂的问题。本文采用人工干预和自动作图结合的方案: 提前绘制几何题的初始图形、解答内容的辅助图形和通过指令接口无法准确绘制的图形(例如角度符号和相等符号), 将这些预置图形保存在一个 GGB 文件中, 在讲读运行时在网页中加载这些预置图形(由 C++应用将 GGB 文件转换为 BASE64 编码串, 传递给 GGB 接口实现加载^[1])并动态创建亮显效果的图形。这样就解决了图形定位问题, 充分利用 GGB 软件的易用性降低几何讲读工具的复杂性。

中学几何题通常有题目和数量不等的解答部分, 对应的图形通常会有差异, 例如解答部分有辅助线。对于动态切换题目和不同解答部分对应的图形, 本文采用如图 9 所列算法动态切换预置图形的显示条件来实现: 对解答部分特有的图形, 设置其显示条件(在图形属性对话框的“高级”页面中)为“group==1”(表示仅在第一个解答部分才显示)或“line>=12”(表示从几何题的第 12 行起才显示), 而对题目内容对应的图形则不设置显示条件, 让其总是显示。“group”和“line”变量可以在 GGB 软件的指令栏中初始化(例如输入“group=1”), 在讲读运行时调用 GGB 接口的 evalCommand 函数动态设置值, 从而实现动态显示。

```
var Hi = window.Hi = window.Hi || { defaultTimeout: 1500 };
Hi.init = function(text) { // 根据几何题文本, 初始化 GGB 亮显的环境参数
    Hi.lines = text.replace(/\n\r|\r\n/g, '\n').split('\n'); // 拆分文本行
    Hi.queue = parseLines(Hi.lines).map(function(r) {
        var timeout = r.rel ? 'Hi.defaultTimeout * 2' : 'Hi.defaultTimeout';
        var s = 'Hi.' + r.name + "(" + r.key + ", {timeout: " + timeout + "})";
        r.show = new Function(s); // 根据谓词构造亮显函数
        return r;
    });
};
Hi.seek = function(line, key) { // 根据行号和关键字亮显特定图形
    for (var i = 0; i < Hi.queue.length; i++) {
        var item = Hi.queue[i];
        if (item.line === line && item.key === key) {
            ggbApplet.evalCommand ('line=' + line);
            ggbApplet.evalCommand ('step=' + (i + 1));
            ggbApplet.evalCommand ('order=' + (item.line * 10 + item.col + 1));
            item.show(); // 调用亮显函数, 例如 Hi.segment ('AB', {timeout:1000})
            break;
        }
    }
};
```

图 9 作图指令映射和亮显定位的算法

本文的动态亮显方案是将 4.3 节的语义理解结果的谓词逻辑描述映射到 GGB 的作图指令（见图 9 所列算法的“Hi.init”动态创建显示函数部分），然后为每个谓词编写同名的 JS 亮显函数，在其中调用 GGB 的接口函数（例如谓词“angle3p”对应的亮显函数名为“Hi.angle3p”，实现见图 10 所示），并为临时创建的图形设置亮显颜色和加粗等属性，通过延时删除图形实现高亮显示效果。例如对表 3 中的谓词逻辑描述采用表 5 所示的作图指令自动作图。其中“ $AB \parallel CD$ ”等二元图形关系的谓词逻辑将映射到多个作图指令，通过设置不同的延时删除时间达到重叠亮显效果，例如让线段 AB 亮显持续到线段 CD 亮显完成后才消失。

```
function addHighLight(name, attr) { // 设置亮显图形的显示属性
  ggbApplet.setLabelVisible(name, false); // 不显示标签, 仅亮显图形
  ggbApplet.setColor(name, 0, 0, 255); // 默认蓝色亮显
  ggbApplet.setFixed(name, true); // 亮显图形不可拖动
  if (attr && attr.timeout) { // 延时删除亮显图形
    setTimeout(function(){ggbApplet.deleteObject(name);}, attr.timeout);
  }
  return name;
}
// 线段亮显函数, 例如调用 Hi.segment('AB', {timeout: 1000})
Hi.segment = function(ab, attr) {
  var hi = ab + '_{his}'; // 下面根据端点创建线段, 要求提前在 GGB 文件中画出端点
  ggbApplet.evalCommand(hi + '=Segment[' + ab[0] + ',' + ab[1] + ']');
  ggbApplet.setLineThickness(hi, 7); // 加粗亮显
  return addHighLight(hi, attr);
};
//  $\angle ABC$  形式的角度亮显函数, 例如调用 Hi.angle3p('PAB', {timeout: 1000})
Hi.angle3p = function(abc, attr) {
  Hi.segment(abc[0] + abc[1], attr); // 亮显第一条边
  Hi.segment(abc[1] + abc[2], attr); // 亮显第二条边
};
```

图 10 亮显函数示例

表 5 自动作图的指令映射示例

输入文本	结论：一阶谓词逻辑	GGB 作图指令或函数
$AB \parallel CD$	segment(AB, //)、 segment(CD)	Segment[A,B]、Segment[C,D]
$AB=CD$	segment(AB, =)、 segment(CD)	
四点 $ABEP$ 共圆	circle3p(ABEP)	Circle[A,B,E]
圆 ABC	circle3p(ABC)	Circle[A,B,C]
$\odot O$	circle(O)	Circle[O, 查到同心的圆的名称]
$\angle PAB$	angle3p(PAB)	Angle[P,A,B]
平行四边形 $ABCD$	polygon(ABCD)	Polygon[A,B,C,D]
CD 的延长线	extension(CD)	Ray[D, Reflect[C,D]]
射线 AB	ray(AB)	Ray[A,B]
AB	segment(AB)	Segment[A,B]

5 结束语

本文针对常见的几何讲题方式的局限性, 提出了基于几何语义理解、GGB 自动作图和语音合成内核(TTS)的几何自动讲题方式, 并基于 HTML5 和 Hybrid 混合架构开发了语音可视化几何讲题工具, 具有语音朗读与图形亮显同步、易于修改和随机回放的特点。

本几何讲题方式适合人工智能时代的互联网教学的趋势, 但在推广上有一定难度。教学中电脑的普及给这种讲题方式奠定了一定的基础, 但仍需要培养大量创新型的老师, 学会制作 GGB 图形。目前本研究采用人工干预和自动作图结合的方案。后期需要深入研究全自动智能作图技术(即根据题目和答案的文本, 按照步骤自动作图), 以便降低制作和推广难度。

参考文献

- [1] 张云贵. GeoGebra 教育资源云平台的构建研究[J]. 中国现代教育装备, 2017(1):19-21.
- [2] 陈肖宇. 电子几何教科书系统的设计与实施[D]. 北京航空航天大学, 2011.
- [3] 洪宇. 将 webkit 内核封装为 duilib 的浏览器控件[EB/OL]. [2014-07-26]. <http://blog.csdn.net/zhuhongshu/article/details/38145651>.
- [4] 黄焕, 刘清堂, 陈矛. 面向平面几何命题理解的融合算法研究[J]. 计算机科学, 2013, 40(8):196-199.
- [5] 郭海燕. 受限域下动态几何自然语言作图方法研究[D]. 华中师范大学, 2012.
- [6] 马创新. 论知识表示[J]. 现代情报, 2014, 34(3):21-24.

本研究报告的参赛队员为何婉榕一人，何婉榕在导师的指导下完成了整个研究报告的撰写。

何婉榕（2001-），女，清华大学附属中学 高一第一创新班学生

本人喜欢运动，唱歌等，乒乓球打得不错。高一期间担任班上的财务委员（初中担任了 3 年的卫生委员）。喜欢参加各项课内外活动，高一一年中，在北京古动物馆做过志愿者讲解员；参加了 12 天的 2017 冬季普林斯顿青少年领袖峰会；成为“清华附中第六届自主课堂”讲师，给大家讲《青年理财》等。

本人兴趣广泛，创新思维能力强，平常有很多想法和点子，爱好科技发明创造，尤其是人工智能方向，高一带领小组并作为主力研发了《电子书语音自动翻页神器》，正在申请一个“发明专利”和一个“实用新型专利”。

本人平时学习努力，综合成绩优秀（在年级第 7 名和第 19 名之间波动），积极参加各项竞赛，并获得了较好的成绩和奖项。

综合：高一：2017 清华附中启迪奖 校长奖（高中部 6 人获此奖，高一 1 人）

数学：高一：2017 年北京市中学生数学竞赛 高一年级 一等奖

初三：2016 全国初中数学联赛 一等奖

物理：高一：第 30 届北京市高中力学竞赛 决赛 一等奖

语文：高一：第 12 届全国中小学生创新作文大赛高中组 全国总决赛 二等奖

英语：高一：第 16 届全国英语创新大赛 决赛 全国一等奖

发明专利（申请中）：自动翻页方法及装置，201710254576.4，何婉榕，

实用新型（申请中）：智能电子终端自动翻页装置，201720406620.4，何婉榕等

周建军老师，男，清华大学附属中学的数学老师，高一学部主任，高一数学备课组长，数学与技术教研组联合党支部书记，海淀区学科带头人。

张云贵老师，男，北京北大方正电子有限公司高级软件工程师，硕士研究生毕业，研究方向为人工智能和教育应用技术，长期研究 GGB 云平台应用技术。

本参赛团队声明所提交的论文是在指导老师指导下进行的研究工作和取得的研究成果。尽本团队所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果。若有不实之处，本人愿意承担一切相关责任。

参赛队员： 何婉榕 指导老师： 周建军、张云贵

2017 年 8 月 30 日