工程技术与管理·第 04 卷·第 02 期·2020 年 02 月
DOI: https://doi.org/10.26549/gcjsygl.v4i2.3417

Two Common Programs for Calculating Middle Pile Elevation in Bridge and Culvert Design

Jinguang Liu[1] Xingguang Liu[1] Di Liu[2] Guangming Tian[1]

1. Tai'an Highway Development Center, Tai'an, Shandong, 27100, China
2. Tai'an Branch of Shandong Highway Design Consulting Co., Ltd., Tai'an, Shandong, 27100, China

Abstract

Bridge and culvert design often require the middle pile height of one or a lot of piles. It is more troublesome for the designer to find the route to get the data. For this situation, two auxiliary programs have been written for the paper. These two programs are very convenient in the design of bridges and culverts, which improves work efficiency and accuracy and avoids errors. At the same time, they have the basic kernel of the route design program. Through the openness of the source code of these two programs, more design colleagues can understand and develop applications suitable for themselves.

Keywords

bridge and culvert design; data; auxiliary program

桥涵设计中计算中桩高程的两个常用程序

刘金光[1] 刘兴光[1] 刘笛[2] 田光明[1]

1. 泰安市公路事业发展中心，中国·山东 泰安 27100
2. 山东省公路设计咨询有限公司泰安分公司，中国·山东 泰安 27100

摘 要

桥涵设计中经常需要某个或批量桩号的中桩高程，找路线设计师索取数据比较麻烦，针对这种情况开编写了论文的两个辅助程序。这两个程序在桥涵设计工作中非常方便，提高了工作效率及精度，避免了错误。同时，它们拥有路线设计程序的基本内核，通过这两个程序源代码的开放，让更多的设计同行们了解和开发适合自己的应用程序。

关键词

桥涵设计；数据；辅助程序

1 引言

自 AutoCAD 应用到设计工作中以来，其强大的功能给设计人员提供了非常高效的帮助，但是，它并不是万能的，针对一些特殊的功能要求它不会提供，只有通过二次开发才能实现，幸好 AutoCAD 提供了这个便利，它能够支持 AutoLISP、ADS、Object ARX、VisuallISP、VBA 等。论文的两个程序就是在 AutoCAD 平台上直接使用的应用工具，采用 AutoLISP 语言编写，对于编写小规模应用工具的设计人员来说，我推荐使用 AutoLISP，其短小灵活，比较容易调试，不需要学习更多的编程知识。

论文介绍的两个程序中 ZD.lsp 是计算一个具体桩号高程的工具，它有一个简单的对话框（见下图），对话框文件为 ZD.dcl，它还需要一个数据文件（文后介绍），数据文件来源于路线设计参数文件；另一个是 ZH.lsp，是在 ZD.lsp 正常计算后可使用的工具，它共享使用 ZD.lsp 的数据文件，它可以将 CAD 可编辑的数字认为是桩号，选择后一次性将其转换成高程数字[1]。（此组程序非常适合从事路线施工放样、桥涵设计的技术人员使用，可作为软件教学的素材。）

2 对话框程序

对话框程序（ZD.dcl，存于 ACAD 可支持的目录下），效果如下：


```
zd : dialog {
        label = /*MSG0*/" 本程序计算全线中桩设计标高
```

DOI: https://doi.org/10.26549/gcjsygl.v4i2.3417


```
（泰安市公路局，作者：刘金光）";
        fixed_width=true;
        initial_focus = "ssta";
         spacer;
        :column {
         spacer;
         fixed_width=true;
        : row{
          : column {
              label = /*MSG1*/" 请输入 :";
             : edit_box {
                      label = /*MSG9*/" 计算桩号 (m)：";
                      key = "ssta";
                      edit_width = 9;
                     }
                }
         spacer;
         spacer;
         :column {
              label = /*MSG51*/" 中桩标高（m）：";
              : text {
                        key = "slev";
                        width = 9;
                    spacer;
                  }
                }
         spacer;
           :row {
          spacer;
          repeat_button;
          spacer;
          getfil_button;
          spacer;
          cancel_button;
          }
        }
        spacer;
         spacer;
        : row{
           : text {
                key = "path";
               width =true;
                  }
             }
         }
        }
```

3 中桩设计标高计算程序（ZD.lsp）

```
(setq lin_pos '(−1 −1))
(defun c:zd (/ done next_d  slev sta ssta fn1 fn2 pathnam1 pathnam2 temm tem  nam2 p palf sig I ptt1 ptt2 pr pt dl mm msg)；；；定义局部变量
;;; 临时函数
 (defun lamd (temm)
   (+ 0.0 temm)
 )
;;; 容错处理
(defun *error* ( msg)
   (alert  " 参数文件数据格式不正确 1\n      请重新选择！ ")
   (getfil)(princ ))
;;;========== OK 子程序将由此开始 ==========
 (defun ok(/ lin_pos1)
   (initget 1)
   (setq ssta (distof (get_tile "ssta")))
   (setq lin_pos (done_dialog 2))
   (new_dialog "zd" lin_dcl "" lin_pos)
   (setq lin_pos1 (done_dialog 1))  )
;;;============ 链接参数文件子程序 ========
(defun getfil(/ fn d d1 tt ttm)
  (setq tt 0 ttm 0 d nil d1 nil)
   (while (= tt 0)
       (setq pathnam2 (getfiled " 选择要链接的竖曲线参数文件 :" "I:\ 纵断数据共享 " "dat" 0))；；；默认搜索位置
```

工程技术与管理·第 04 卷·第 02 期·2020 年 02 月
DOI: https://doi.org/10.26549/gcjsygl.v4i2.3417

```
;        (setq pathnam2 (getfiled " 选择要链接的竖曲线参数文件 :" (getvar "DWGPREFIX") "dat" 0)); ; ; 提取工作文件路径
            (setq fn (open pathnam2 "r"))
            (setq tt 1)
            (while (and (setq d (read-line fn)) (= ttm 0))
                (if (= nil (listp (read d)))
                    (progn
                        (princ d)
                            (alert  " 参数文件数据格式不正确 2\n 请重新选择!  ")
                        (setq tt 0 ttm 1)
                    ))))
            (setq nam2 (strcat " 当前链接的竖曲线参数文件为：" pathnam2))
        (wfile)(reading)(ok)(close fn))
    ;;;============ 将参数文件链接路径保存子程序 ==============
    (defun wfile(/ fn)
        (setq fn (open pathnam1 "w"))
        (princ pathnam2 fn)
        (close fn))
    ;;;======= 读参数文件或链接路径子程序 ========
    (defun reading(/ K kl fai1 fai2 p1 alf tt r t1 t2)
      (setq p nil p1 nil dl nil)
      (setq pathnam1 "c:/pathnam.path")
      (if (not (setq fn1 (open pathnam1 "r")))
          (getfil)
          (progn
            (setq pathnam2 (read-line fn1))
            (if pathnam2 (princ) (getfil))
          ))
      (setq fn2 (open pathnam2 "r"))
      (setq dl (read-line fn2))
      (setq dl (read dl))
      (setq kl 0)
      (while (setq p1 (read-line fn2))
          (setq p1 (read p1))
          (setq p1 (mapcar 'lamd p1))
          (while (and (nth kl dl) (> (car p1) (nth kl dl)))
              (setq t1 (+ (car p1) (nth (+ kl 1) dl)))
              (setq t2 (cdr p1))
              (setq t2 (reverse t2))
              (setq p1 (append t2 (list t1)))
              (setq p1 (reverse p1))
              (setq kl (+ 2 kl)))
          (setq p (append p p1))
          (setq kl 0))
      (close fn2)
    ;;; Palf 转角计算
      (setq palf nil sig nil I 0 K 0 fai1 0 fai2 0 alf 0)
      (while (nth (+ (* 3 I) 3) p)
          (if (= I 0) (setq palf '(0) sig '(0) I 1))
          (setq K (* I 3))
          (setq fai1 (/ (- (nth (+ k 1) p) (nth (- k 2) p)) (- (nth (+ k 0) p) (nth (- k 3) p))))
          (setq fai2 (/ (- (nth (+ k 4) p) (nth (+ k 1) p)) (- (nth (+ k 3) p) (nth (+ k 0) p))))
          (setq temp (- fai2 fai1))
          (setq alf (abs temp))
          (setq alf (list alf))
          (setq temp (list temp))
          (setq palf (append palf alf))
          (setq sig (append sig temp))
          (setq I (+ I 1))
      )
      (setq palf (append palf (list 0)))
      (setq sig (append sig (list 0)))
    ;;; 切线长 PT、切点桩号 PTT 及半径 PR
      (setq pt nil ptt1 nil ptt2 nil pr nil I 0 K 0)
      (while (nth (+ (* 3 I) 3) p)
          (if (= I 0) (setq  pt '(0) ptt1 '(0)  ptt2 '(0) pr '(0) I 1))
          (setq K (* I 3))
          (if (> (nth (+ k 2) p) 0)
```

```
                (progn
                  (setq pr (append pr (list (nth (+ k 2) p))))
                    (setq tt (* (nth (+ k 2) p) (/ (sin (* 0.5 (nth I
palf))) (cos (* 0.5 (nth I palf))))))
                  (setq pt (append pt (list tt)))
                  (setq ptt1 (append ptt1 (list (- (nth k p) tt))))
                  (setq ptt2 (append ptt2 (list (+ (nth k p) tt))))
                )
                (progn
                  (setq tt (abs (nth (+ k 2) p)))
                  (setq pt  (append pt (list tt)))
                   (setq r (/ tt (/ (sin (* 0.5 (nth I palf))) (cos (* 0.5
(nth I palf))))))
                  (setq pr (append pr (list r)))
                  (setq ptt1 (append ptt1 (list (- (nth k p) tt))))
                  (setq ptt2 (append ptt2 (list (+ (nth k p) tt))))
                )
              )
              (setq I (+ I 1))
            )
            (setq pr (append pr (list 0)))
            (setq ptt1 (append ptt1 (list (nth (+ k 3) p))))
            (setq ptt2 (append ptt2 (list (nth (+ k 3) p))))
            (setq nam2 (strcat " 当前链接的竖曲线参数文件为：" pathnam2))
            (set_tile "path" nam2)
          )
          ;;;=========== JISUAN 计算子程序 ===========
          (defun jisuan (/ II jk tem kk slevel dh)
            (setq II 0 kk 0 jk 0 tem 0.0 slevel 0.0 dh 0.0)
            (if (or (< sta (nth 0 p)) (> sta (nth (* 3 I) p))) (alert " 桩号有误！请重输。"))
            (while (and (nth (+ (* 3 II) 3) p) (= 0 jk))
              (setq kk (* 3 II))
              (while (and (>= sta (nth kk p)) (< sta (nth (+ kk 3) p))
(= 0 jk))
                (if (and (>= sta (nth kk p)) (< sta (nth II ptt2)))
                  (if (= 0 (nth II pr))
                    (setq dh 0.0)
                    (progn
                      (setq dh (/ (expt (- (nth II ptt2) sta) 2) (*
2 (nth II pr))))
                      (if (> 0 (nth II sig)) (setq dh (* dh -1.0)))
                    )
                  )
                )
                (if (and (>= sta (nth II ptt2)) (<= sta (nth (+ II 1)
ptt1)))
                  (setq dh 0.0 )
                )
                (if (and (> sta (nth (+ II 1) ptt1)) (< sta (nth (+ kk
3) p)))
                  (if (= 0 (nth (+ II 1) pr))
                    (setq dh 0.0 )
                    (progn
                      (setq dh (/ (expt (- (nth (+ II 1) ptt1) sta) 2)
(* 2 (nth (+ II 1) pr))))
                      (if (> 0 (nth (+ II 1) sig)) (setq dh (* dh
-1.0)))
                    )
                  )
                )
                (setq tem (/ (- (nth (+ kk 4) p) (nth (+ kk 1) p)) (-
(nth (+ kk 3) p) (nth kk p))))
                (setq slevel (+ dh (nth (+ kk 1) p) (* tem (- sta (nth
kk p)))))
                (setq jk 2)
              )
              (setq II (+ II 1))
            )
            (setq slevel slevel)
          )
          ;;;========== DLK 断链处理子程序 ===========
          (defun dlk (/ kl km ssta1 ssta2 slev1 slev2 slev3 tem)
```

工程技术与管理·第 04 卷·第 02 期·2020 年 02 月
DOI: https://doi.org/10.26549/gcjsygl.v4i2.3417

```
(setq kl 0 km 0 slev1 0.0 slev2 nil slev3 nil tem 0.0)
(setq ssta1 ssta ssta2 ssta)
(while (and (nth kl dl) (> ssta (nth kl dl)))
    (setq ssta1 (+ ssta1 (nth (+ kl 1) dl)))
    (setq tem (- ssta1 ssta))
    (setq km kl)
    (setq kl (+ 2 kl))
)
(if (and (nth kl dl) (> (nth (+ kl 1) dl) 0) (>= ssta2 (- (nth kl dl) (nth (+ kl 1) dl))) (<= ssta2 (nth kl dl)))
    (progn
      (setq ssta2 (+ ssta2 tem (nth (+ kl 1) dl)))
      (setq sta ssta2)
      (setq slev2 (rtos (jisuan) 2 3))
    )
)
(if (and (nth km dl) (< (nth (+ km 1) dl) 0) (> ssta (nth km dl)) (< ssta (- (nth km dl) (nth (+ km 1) dl))))
    (setq slev3 " 该桩号不存在 !")
)
(setq sta ssta1)
(setq slev1 (rtos (jisuan) 2 3))
(if slev2 (setq slev1 (strcat slev1 " / " slev2)))
(if slev3 (setq slev1 slev3))
(setq slev1 slev1)
)

;;;========== 主程序由此开始 ==========
(setq slev 0.0 ssta 0.0 mm 0 p nil next_d 5)；；；预设初始量
(reading)
(setq nam2 (strcat " 当前链接的竖曲线参数文件为：" pathnam2))
(setq lin_dcl (load_dialog "zd.dcl"))；；；调取对话框程序
(if (< lin_dcl 0) (exit))；；；调取对话框程序失败退出
(while (< 1 next_d)
(if (not (new_dialog "zd" lin_dcl "" lin_pos)) (exit))；；；找不到对话框退出
(set_tile "slev" (rtos slev 2 3))；；；标高数值转换为字符串
(set_tile "ssta" (rtos ssta 2 3))；；；桩号数值转换为字符串
(action_tile "repeat" "(ok)")；；；确认后计算
(if (> mm 0) (set_tile "slev" (dlk)))；；；调取断链子程序
(set_tile "path" nam2)
(action_tile "getfil" "(getfil)")
(action_tile "cancel" "(wfile)(done_dialog 0)(setq next_d 0)")；；；取消后重置
(setq next_d (start_dialog))
(setq mm (+ mm 1))；；；循环计数器
)
(princ "\n  欢迎使用本全线标高计算程序！ ")
(princ)
)
```

4 桩号转换为中桩设计标高程序（ZH.lsp）

```
(defun c:ZH (/ done next_d  slev sta ssta fn1 fn2 pathnam1 pathnam2 temm tem temn chm
             nam2 p palf sig I ptt1 ptt2 pr pt dl msg txt jk kk s sy sd e Nn as *error*)
;;; 临时函数
(defun lamd (temm)
  (+ 0.0 temm))
;;; 容错处理
(defun *error* ( msg)
  (alert  " 请调用 ZD 程序，检查纵断参数文件是否有错！ ")
  (princ ))
;;;========== 链接参文件子程序 ==========
(defun getfil(/ fn d d1 tt ttm)
  (setq tt 0 ttm 0 d nil d1 nil)
  (while (= tt 0)
```

```
        (setq pathnam2 (getfiled " 选择要链接的竖曲线参数文件 :" (getvar "DWGPREFIX") "dat" 0))
    ;       (setq pathnam2 (getfiled " 选择要链接的竖曲线参数文件 :" "//SJS_ 服务器 /F/ 纵断数据共享 /" "path" 0))
        (setq fn (open pathnam2 "r"))
        (setq tt 1)
        (while (and (setq d (read-line fn)) (= ttm 0))
            (if (= nil (listp (read d)))
                (progn
                    (alert  " 参数文件数据格式不正确 \n    请重新选择！ ")
                    (setq tt 0 ttm 1)
    ))))
        (close fn)(wfile)(princ)
    )
    ;;;======== 将参数文件链接路径保存子程序 ======
    (defun wfile(/ fn)
        (setq fn (open pathnam1 "w"))
        (princ pathnam2 fn)
        (close fn))
    ;;;============ 读参数文件或链接路径子程序 ==============
    (defun reading(/ K kl fai1 fai2 p1 alf tt r t1 t2)
      (setq p nil p1 nil dl nil)
      (setq pathnam1 "c:/pathnam.path")
      (if (not (setq fn1 (open pathnam1 "r")))
          (getfil)
          (progn
            (setq pathnam2 (read-line fn1))
            (if pathnam2 (princ) (getfil))
          ))
      (setq fn2 (open pathnam2 "r"))
      (setq dl (read-line fn2))
      (setq dl (read dl))
      (setq kl 0)
      (while (setq p1 (read-line fn2))
          (setq p1 (read p1))
          (setq p1 (mapcar 'lamd p1))
          (while (and (nth kl dl) (> (car p1) (nth kl dl)))
              (setq t1 (+ (car p1) (nth (+ kl 1) dl)))
              (setq t2 (cdr p1))
              (setq t2 (reverse t2))
              (setq p1 (append t2 (list t1)))
              (setq p1 (reverse p1))
              (setq kl (+ 2 kl))
          )
          (setq p (append p p1))
          (setq kl 0))
      (close fn2)
    ;;; Palf 转角计算
      (setq palf nil sig nil I 0 K 0 fai1 0 fai2 0 alf 0)
      (while (nth (+ (* 3 I) 3) p)
          (if (= I 0) (setq palf '(0) sig '(0) I 1))
          (setq K (* I 3))
          (setq fai1 (/ (- (nth (+ k 1) p) (nth (- k 2) p)) (- (nth (+ k 0) p) (nth (- k 3) p))))
          (setq fai2 (/ (- (nth (+ k 4) p) (nth (+ k 1) p)) (- (nth (+ k 3) p) (nth (+ k 0) p))))
          (setq temp (- fai2 fai1))
          (setq alf (abs temp))
          (setq alf (list alf))
          (setq temp (list temp))
          (setq palf (append palf alf))
          (setq sig (append sig temp))
          (setq I (+ I 1))
      )
      (setq palf (append palf (list 0)))
      (setq sig (append sig (list 0)))
    ;;; 切线长 PT、切点桩号 PTT 及半径 PR
      (setq pt nil ptt1 nil ptt2 nil pr nil I 0 K 0)
      (while (nth (+ (* 3 I) 3) p)
          (if (= I 0) (setq  pt '(0) ptt1 '(0)  ptt2 '(0) pr '(0) I 1))
          (setq K (* I 3))
          (if (> (nth (+ k 2) p) 0)
```

工程技术与管理·第 04 卷·第 02 期·2020 年 02 月
DOI: https://doi.org/10.26549/gcjsygl.v4i2.3417

```
                (progn
                  (setq pr (append pr (list (nth (+ k 2) p))))
                    (setq tt (* (nth (+ k 2) p) (/ (sin (* 0.5 (nth I palf))) (cos (* 0.5 (nth I palf))))))
                  (setq pt (append pt (list tt)))
                  (setq ptt1 (append ptt1 (list (- (nth k p) tt))))
                  (setq ptt2 (append ptt2 (list (+ (nth k p) tt))))
                )
                (progn
                  (setq tt (abs (nth (+ k 2) p)))
                  (setq pt  (append pt (list tt)))
                   (setq r (/ tt (/ (sin (* 0.5 (nth I palf))) (cos (* 0.5 (nth I palf))))))
                  (setq pr (append pr (list r)))
                  (setq ptt1 (append ptt1 (list (- (nth k p) tt))))
                  (setq ptt2 (append ptt2 (list (+ (nth k p) tt))))
                ))
              (setq I (+ I 1)))
        (setq pr (append pr (list 0)))
        (setq ptt1 (append ptt1 (list (nth (+ k 3) p))))
        (setq ptt2 (append ptt2 (list (nth (+ k 3) p))))
        (setq nam2 (strcat " 当前链接的竖曲线参数文件为：" pathnam2)))
      ;;;=========== DLK 断链处理子程序 ==========
      (defun dlk (/ kl km ssta1 ssta2 slev1 slev2 slev3 tem)
        (setq kl 0 km 0 slev1 0.0 slev2 nil slev3 nil tem 0.0)
        (setq ssta1 ssta ssta2 ssta)
        (while (and (nth kl dl) (> ssta (nth kl dl)))
            (setq ssta1 (+ ssta1 (nth (+ kl 1) dl)))
            (setq tem (- ssta1 ssta))
          (setq km kl)
            (setq kl (+ 2 kl)))
        (if (and (nth kl dl) (> (nth (+ kl 1) dl) 0) (>= ssta2 (- (nth kl dl) (nth (+ kl 1) dl))) (<= ssta2 (nth kl dl)))
            (progn
              (setq ssta2 (+ ssta2 tem (nth (+ kl 1) dl)))
              (setq sta ssta2)
              (setq slev2 (rtos (jisuan) 2 3))
            ))
        (if (and (nth km dl) (< (nth (+ km 1) dl) 0) (> ssta (nth km dl)) (< ssta (- (nth km dl) (nth (+ km 1) dl))))(setq slev3 " 该桩号不存在 !"))
        (setq sta ssta1)
        (setq slev1 (rtos (jisuan) 2 3))
        (if slev2 (setq slev1 (strcat slev1 "/" slev2)))
        (if slev3 (setq slev1 slev3))
        (setq slev1 slev1))
      ;;;=========== 主程序由此开始 ==========
      (setq CVAR (getvar "CMDECHO"))
      (setvar "CMDECHO" 0)
      (setq slev 0.0 ssta 0.0  p nil chm 0)
      (reading)
      (prompt "\n  选择要将桩号转换成标高的数字 ...")
      (setq txt (ssget))
      (initget 1)
      (setq kk 0  n (sslength txt))
      (repeat n
          (setq temn (cdr (assoc 0 (setq e (entget (ssname txt kk))))))
        (if (= "TEXT" temn)
          (progn
            (setq s (cdr (setq as (assoc 1 e)))
                sy (atof s)
                ssta sy
                sy (dlk)
                sd (cons 1 sy)
                 e (subst sd as e))
            (entmod  e)
            (setq chm (1+ chm))
          ))
        (setq kk (+ 1 kk))
      )
      (setvar "CMDECHO" CVAR)
      (princ "Changed ")
```

(princ chm)

(princ " text lines. 用 ZD 程序验证并改变参数文件 .")

(setq nam2 (strcat " 当前链接的竖曲线参数文件为：" pathnam2))

(princ nam2)

(princ " 欢迎使用桩号转换标高程序！ ")(princ))

;;;========== ZH 主程序结束 ==========

5 纵断面设计参数填写格式说明：

；后缀 dat，单位：米。数据从路线软件（如纬地）纵断面设计参数文件中提取

；编辑格式如下：

；第一行为断链参数必填，没有断链填（0 0），断链起点，断链长度（长链为负值）

；第二行，起点桩号，高程 竖曲线半径

；第三行，第二个竖曲线，桩号 高程 竖曲线半径（负值为切线长度）

；其它行类推

；最后一行，终点，桩号 高程 竖曲线半径

数据如下：

(3738.389 −261.611)

(2200 138.54 0)

(2420 138.9 −43.06)

(3200 133.440 −175.57)

(3464 135.1 −88.43)

(3655 138.15 −102.00)

(4170 130.3 −140)

…

(12278.385 154.87 0)

6 结语

为便于学习使用，论文所涉及的程序及参数文件均在百度网盘中共享：

源代码下载地址链接：

https://pan.baidu.com/s/1x7ySw0bzgidgr3TgpPPHwQ

提取码：neex

参考文献

[1] 郭平平 , 梁凡 .AutoLISP R13&DCL 从入门到精通