

## **Rutine pentru reprezentarea filetelor în AUTOCAD**

**Maria Oltean**

**UNIVERSITATEA DIN ORADEA**

### **Abstract**

AUTOCAD is probably the most representative CAD program. Besides its simplify and accessibility AUTOCAD developed a mechanism to link with other programs (AUTOLISP, C++, VB, EXCEL, WORD), enhancing its own performances. The work introduces three new routines developed in AUTOLISP by adding new commands and customizing AUTOCAD. These commands allow the easy drawing of the threading hole symbols, in front and side view in the mechanical design.

### **1. Introducere**

În proiectarea asistată de calculator, programul AUTOCAD ocupa un loc de referință. Aceasta pentru că, fiind unul dintre primele programe de proiectare asistată, prin generalitatea, simplitatea și nevoia redusă de resurse, cât și prin prețul său accesibil, a devenit în timp cel mai cunoscut, mai răspândit și mai folosit program. Desigur, față de alte programe mai sofisticate precum CATIA, UNIGRAPHIX, SOLIDWORKS, posibilitățile sale, mai ales pentru anumite domenii cu cerințe speciale, sunt mai reduse, însă pentru majoritatea aplicațiilor AUTOCAD este soluția optimă. Pentru a putea răspunde unui număr cât mai mare de aplicații, cât și pentru a face față cerințelor mereu în creștere, AUTOCAD, în afara faptului că lansează periodic versiuni noi sau particularizate, a dezvoltat un mecanism de interfatare cu alte programe precum AUTOLISP, C++, EXCEL, WORD, VB, diferite programe pentru baze de date, folosind posibilitățile acestora pentru a completa și dezvolta posibilitățile intrinseci ale programului.

Programul AUTOLISP permite dezvoltarea unor rutine specifice care introduse sub forma unor comenzi noi particularizează programul AUTOCAD adăugându-i noi valențe și îmbunătățindu-i performanțele.

În proiectarea mecanica, tehnica de proiectare folosind programul AUTOCAD a evoluat de la simpla reprezentare a elementelor grafice ale pieselor, asemanatoare desenarii lor pe planseta în proiectarea clasica, la tehnica modelarii. Aceasta presupune realizarea în spatiul model (**Model space**) a unui model spatial al piesei, care este apoi vizualizata în spatiul hîrtie (**Paper space**) sub forma unor proiectii minim necesare, pentru evidentierea tuturor detaliilor de executie. Pe aceste vederi se realizeaza cotarea, avînd acces la punctele critice ale modelului, si se deseneaza sau insereaza alte elemente grafice ale proiectului (axe, simboluri de gauri si filete, rugozitati, note). În acest context este foarte important modul în care se introduc simbolurile pentru reprezentarea gaurilor si a gaurilor filetate, întrucît aceste elemente caracteristice îmbinarilor cu filet sunt foarte des utilizate si este important ca reprezentarea sa fie facuta prin cît mai putine comenzi si mai putin consumatoare de timp. Acest lucru se poate rezolva în mod optim prin dezvoltarea si utilizarea unor rutine AUTOLISP.

## 2. Rutina AUTOLISP pentru marcarea centrului gaurii filetate

Rutina marcheaza centrul oricarei gauri în vedere frontala inserînd simbolul respectiv pe un layer dedicat si cu culoarea si tipul de linie stabilite ale acestuia. Selectarea gaurii se face printr-un click cu mouse-ul pe circumferinta cercului gaurii respective. Pentru ca rutina sa lucreze corespunzator este necesar sa existe, sau sa fie creat în prealabil layerul cu numele "CL" pe care se vor crea elementele grafice ale simbolului.

```
(defun c:dfcen()
(setq cmd (getvar "cmdecho"))
(setvar "cmdecho" 0)
(setq cl (getvar "clayer"))
(setq osm (getvar "osmode"))
(setvar "osmode" 0)
(setq p1 (osnap (getpoint "\nSelectati gaura") "nea"))
(setq p2 (osnap p1 "cen"))
(setq p3 (polar p2 0 (* (distance p1 p2) 1.25)))
(setq p4 (polar p2 (/ pi 2) (* (distance p1 p2) 1.25)))
(setq p5 (polar p2 pi (* (distance p1 p2) 1.25)))
(setq p6 (polar p2 (* 3 (/ pi 2)) (* (distance p1 p2) 1.25)))
(setvar "clayer" "cl")
(command "color" "bylayer")
(setq d (/ (distance p2 p3) 7))
(command "line" p2 (polar p2 0 d) "")
(command "line" (polar p2 0 (* 2 d)) p3 "")
(command "line" p2 (polar p2 (/ pi 2) d) "")
(command "line" (polar p2 (/ pi 2) (* 2 d)) p4 "")
(command "line" p2 (polar p2 pi d) "")
(command "line" (polar p2 pi (* 2 d)) p5 "")
(command "line" p2 (polar p2 (* 3 (/ pi 2)) d) "")
(command "line" (polar p2 (* 3 (/ pi 2)) (* 2 d)) p6 "")
(command "color" "bylayer")
```

```
(setvar "cmdecho" cmd)
(setvar "clayer" cl)
(setvar "osmode" osm)
(princ)
)
```

### 3. Rutina AUTOLISP pentru reprezentarea filetelui în vederea frontala a acestuia.

Rutina insereaza simbolul de marcarea centrului gaurii prin apelarea rutinei precedente "defcen" si în completare a simbolului de filet. Pentru aceasta rutina solicita factorul de scara al viewportului pe care îl obtine în urma unui clic pe conturul acestuia si diametrul gaurii printr-un clic pe circumferinta cercului. În cazul în care diametrul gaurii nu corespunde cu diametrul de vârf stabilit pentru fiecare dintre filetele normalizate rutina se întrerupe si utilizatorul este avertizat ca gaura respectiva nu este gaura de filet, altfel este desenat simbolul de filet cu diametrul egal cu diametrul normalizat al filetelui ajustat cu factorul de scara al viewportului. Prin cotarea ulterioara a arcului astfel desenat se obtine valoarea nominala a diametrului filetelui. Elementele filetelui sunt de asemenea desenate pe acelasi layer "CL" a carui existenta este necesara pentru functionarea rutinei.

```
(defun c:filfront()
(setq cmd (getvar "cmdecho"))
(setvar "cmdecho" 0)
(setq osm (getvar "osmode"))
(setvar "osmode" 0)
(setq cl (getvar "clayer"))
(setq e (entsel))
(setq vp (car e))
(setq pvp (cadr e))
(setq evp (entget vp))
(while (/= (cdr (assoc 0 evp)) "VIEWPORT")
(c:filfront)
)
(command "dim" "dimlfac" "v" pvp "exit")
(setq fs (abs (getvar "dimlfac")))
(c:defcen)
(setq lt (getvar "celtype"))
(setq dd (distance p1 p2))
(setq dd1 (* fs dd))
(setq dd1 (rtos dd1 2 2))
(setq dd1 (atof dd1))
(setvar "clayer" "cl")
(command "color" "3")
(setvar "celtype" "continuous")
(cond ((= dd1 2.1) (setq d (/ 2.5 fs)))
(= dd1 2.5) (setq d (/ 3 fs)))
(= dd1 3.4) (setq d (/ 4 fs)))
(= dd1 4.25) (setq d (/ 5 fs)))
```

```

(= dd1 5.1) (setq d (/ 6 fs)))
(= dd1 7) (setq d (/ 8 fs)))
(setq d 0)
)
(if (/= d 0)
(command "arc" (polar p2 (/ (* -5 pi) 180) d) "c" p2 (polar p2 (/ (* 275 pi) 180) d))
;(command "circle" p2 d)
(alert "Nu este gaura de filet!")
)
(command "color" "bylayer")
(setvar "cmdecho" cmd)
(setvar "clayer" cl)
(setvar "osmode" osm)
(setvar "celtype" lt)
(princ)
)

```

#### 4. Rutina AUTOLISP pentru reprezentarea filetului în vederea longitudinala a acestuia.

Rutina creaza si insereaza linia de fund de filet precum si celelalte elemente grafice a filetului în vedere longitudinala. La lansarea comenzii rutina solicita introducerea locatiei centrului gaurii, a razei si a adâncimii gaurii, toate acestea furnizându-i-se prin indicarea cu mouse-ul a punctelor caracteristice în vederea longitudinala a acesteia. Se mai solicita în plus informatia daca gaura este strapunsa sau înfundata, in acest din urma caz linia de marcare a limitei utile a filetului fiind trasata la 5mm de linia ce delimiteaza fundul gaurii. La fel ca si în rutina precedenta, diametrul gaurii trebuie sa corespunda cu diametrul de vîrf al gaurii filetului normalizat pentru a permite inserarea simbolului, in caz contrar aparînd acelasi mesaj de avertizare iar layerul "CL" trebuie sa fie activ.

```

(defun c:fillong();
(setq cmd (getvar "cmdecho"));
(setvar "cmdecho" 0)
(setq osm (getvar "osmode"))
(setvar "osmode" 0)
(setq cl (getvar "clayer"))
(command "ucs" "w")
(setq p1 (osnap (getpoint "\nSelectati centrul gaurii") "cen"))
(setq px (car p1))
(setq py (cadr p1))
(setq ssvp (ssget "X" (list (cons 0 "VIEWPORT"))))
(if ssvp
(setq lssvp (sslength ssvp))
)
(if lssvp
(progn
(setq i 0)

```

```

(while (<= i (1- lssvp))
  (setq idvp (cdr (assoc 68 (entget (ssname ssvp i)))))
  (if (/= idvp 1)
    (progn
      (setq pp (cdr (assoc 10 (entget (ssname ssvp i)))))
      (setq dx (cdr (assoc 40 (entget (ssname ssvp i)))))
      (setq dy (cdr (assoc 41 (entget (ssname ssvp i)))))
      (setq p1x (- (car pp) (/ dx 2)))
      (setq p1y (- (cadr pp) (/ dy 2)))
      (setq p2x (+ (car pp) (/ dx 2)))
      (setq p2y (+ (cadr pp) (/ dy 2)))
      (if (and (>= px p1x) (<= px p2x) (>= py p1y) (<= py p2y))
        (progn
          (setq id (cdr (assoc 69 (entget (ssname ssvp i)))))
          (setq h dy)
          )
        )
      )
      )
      )
      )
      (setq i (1+ i))
      )
      )
      )
      (command "mspace")
      (command "mview" "on" "all" "")
      (setvar "cvport" id)
      (setq vh (getvar "viewsize"))
      (command "pspace")
      (setq fs (/ vh h))
      (setq p2 (osnap (getpoint "\nSelectati raza gaurii") "qua"))
      (setq p3 (osnap (getpoint "\nSelectati lungimea gaurii") "cen"))
      (setq osm (getvar "osmode"))
      (setq cl (getvar "clayer"))
      (command "color" "red")
      (setvar "osmode" 4)
      (command "line" p1 p3 "")
      (command "lengthen" "p" 110 p1 p3 "")
      (setvar "clayer" cl)
      (setvar "osmode" osm)
      (command "color" "bylayer")
      (setq dd (distance p1 p2))
      (setq dd1 (* fs dd))
      (setq dd1 (rtos dd1 2 2))
      (setq dd1 (atof dd1))
      (setvar "clayer" "cl")
      (cond ((= dd1 2.5) (setq d (/ 3 fs)))
            ((= dd1 3.4) (setq d (/ 4 fs)))
      )
    )
  )
)

```

```

((= dd1 4.25) (setq d (/ 5 fs)))
((= dd1 5.1) (setq d (/ 6 fs)))
((= dd1 7) (setq d (/ 8 fs)))
((setq d 0))
)
(initget 1 "T I")
(setq ras (getkword "\nGaura de tercere sau infundat (T sau I)?"))
(if (and (= (strcase ras) "T") (/= d 0))
(progn
(command "color" "3")
(command "-linetype" "s" "hidden2" "")
(if (= (car p1) (car p2))
(progn
(setq pp1 (list (car p1) (+ (cadr p1) d) 0))
(setq pp2 (list (car p3) (+ (cadr p3) d) 0))
(setq pp3 (list (car p3) (- (cadr p3) d) 0))
(setq pp4 (list (car p1) (- (cadr p1) d) 0))
(command "line" pp1 pp2 pp3 pp4 "")
(command "-linetype" "s" "bylayer" "")
)
; end of progn
)
; end of if
(if (= (cadr p1) (cadr p2))
(progn
(setq pp1 (list (+ (car p1) d) (cadr p1) 0))
(setq pp2 (list (+ (car p3) d) (cadr p3) 0))
(setq pp3 (list (- (car p3) d) (cadr p3) 0))
(setq pp4 (list (- (car p1) d) (cadr p1) 0))
(command "line" pp1 pp2 pp3 pp4 "")
(command "-linetype" "s" "bylayer" "")
)
; end of progn
)
; end of if
)
; end of progn
)
; end of if
(if (and (= (strcase ras) "I") (/= d 0))
(progn
(command "color" "3")
(command "-linetype" "s" "hidden2" "")
(if (and (= (car p1) (car p2)) (> (car p3) (car p1)))
(progn
(setq pp1 (list (car p1) (+ (cadr p1) d) 0))
(setq pp2 (list (- (car p3) (/ 4 fs)) (+ (cadr p3) d) 0))

```

```
(setq pp3 (list (- (car p3) (/ 4 fs)) (- (cadr p3) d) 0))
(setq pp4 (list (car p1) (- (cadr p1) d) 0))
(command "line" pp1 pp2 pp3 pp4 "")
(command "-linetype" "s" "bylayer" "")
)
; end of progn
)
; end of if
(if (and (= (car p1) (car p2)) (< (car p3) (car p1)))
(progn
(setq pp1 (list (car p1) (+ (cadr p1) d) 0))
(setq pp2 (list (+ (car p3) (/ 4 fs)) (+ (cadr p3) d) 0))
(setq pp3 (list (+ (car p3) (/ 4 fs)) (- (cadr p3) d) 0))
(setq pp4 (list (car p1) (- (cadr p1) d) 0))
(command "line" pp1 pp2 pp3 pp4 "")
(command "-linetype" "s" "bylayer" "")
)
; end of progn
)
; end of if
(if (and (= (cadr p1) (cadr p2)) (< (cadr p3) (cadr p1)))
(progn
(setq pp1 (list (+ (car p1) d) (cadr p1) 0))
(setq pp2 (list (+ (car p3) d) (+ (cadr p3) (/ 4 fs)) 0))
(setq pp3 (list (- (car p3) d) (+ (cadr p3) (/ 4 fs)) 0))
(setq pp4 (list (- (car p1) d) (cadr p1) 0))
(command "line" pp1 pp2 pp3 pp4 "")
)
; end of progn
)
; end of if
(if (and (= (cadr p1) (cadr p2)) (> (cadr p3) (cadr p1)))
(progn
(setq pp1 (list (+ (car p1) d) (cadr p1) 0))
(setq pp2 (list (+ (car p3) d) (- (cadr p3) (/ 4 fs)) 0))
(setq pp3 (list (- (car p3) d) (- (cadr p3) (/ 4 fs)) 0))
(setq pp4 (list (- (car p1) d) (cadr p1) 0))
(command "line" pp1 pp2 pp3 pp4 "")
(command "-linetype" "s" "bylayer" "")
)
; end of progn
)
; end of if
)
; end of progn
)
; end of if
```

```
(if (= d 0)
(alert "Nu este gaura filetata!")
)
(command "color" "bylayer")
;(setvar "cmdecho" cmd)
(setvar "clayer" cl)
(setvar "osmode" osm)
(princ)
)
```

## 5. Concluzii

Rutinele AUTOLISP prezentate se pot ușor modifica și adapta necesităților utilizatorului. Folosirea lor reduce efortul de proiectare crește acuratețea și reduce semnificativ timpul de proiectare. Ele sunt utilizate pentru reprezentarea în spațiul hârtie (**Paper space**), în vedere frontală și longitudinală, a gaurilor filetate (M5, M6, M8, M10) modelate în 3D ca și gauri simple, cu diametrul egal cu diametrul de vîrf al filetului normalizat.

## Bibliografie

1. G. Albeanu : Modelare geometrică pentru grafica și proiectarea asistată de calculator, Editura Tehnica, București, 2001.
2. D. Dragomir : Proiectarea asistată de calculator pentru inginerie mecanică, Teora, 2000.
3. I. Simion : AutoCAD 2002 pentru ingineri, Teora 2003.
4. T. Tiuca, T. Precup, T. Antal : Dezvoltarea aplicațiilor cu AutoCAD și AutoLISP, Editura Promedia Plus Computers, Cluj – Napoca, 1995.
5. CAD Report: <http://www.cadreport.ro>, Agora, Târgu Mureș