

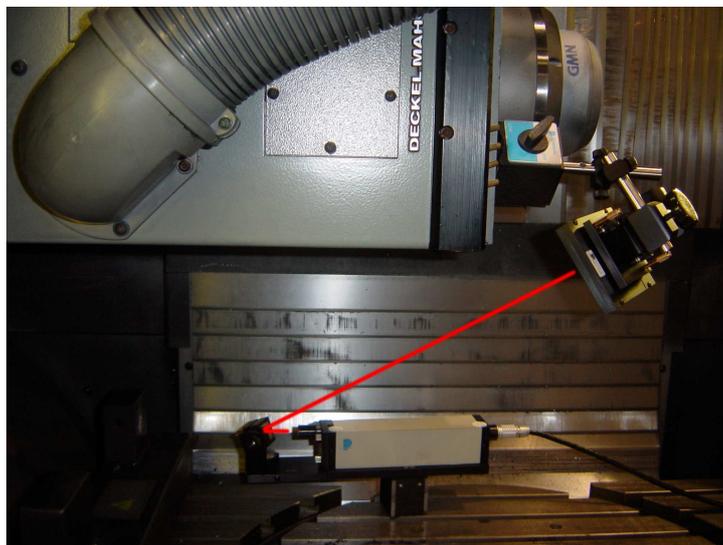
Compensazione Volumetrica 3D sul Part Program (programma pezzo) con interpolazione sia lineare che circolare

I. Introduzione

Per molti controlli, è possibile compensare solo gli errori lineari di ogni asse. Con il programma LDDM la compensazione volumetrica 3D degli errori di posizionamento può essere ottenuta compensando il programma pezzo (codice ISO). Per prima cosa introdurre gli errori di posizionamento volumetrico 3D che sono stati misurati in precedenza, poi inserire il programma pezzo (codice ISO) da compensare, il programma LDDM genererà un programma pezzo compensato.

Questo semplice programma legge solo : G0 per il posizionamento, G1 per interpolazione lineare, G2 (orario) e G3 (antiorario) per l'interpolazione circolare, G17 (piano XY), G18 (piano ZX), G19 (piano YZ) per il piano del percorso circolare e I (centro del percorso circolare in X), J (centro del percorso circolare in Y), K (centro del percorso circolare in Z) e anche G90 (assoluto) e G91 (incrementale). Tutti gli altri comandi non vengono letti e cambiati. Inoltre le linee del programma compensato saranno molte di più di quelle del codice originale.

Sia per l'interpolazione lineare che circolare, il programma divide il percorso utensile in molti segmenti. Utilizzando le informazioni della compensazione volumetrica 3D sugli errori di posizionamento, viene compensato ogni punto di partenza e di arrivo di ogni segmento. Per ottenere una maggiore precisione o con macchine con errori elevati è opportuno dividere in un numero maggiore di segmenti. Per generare una tabella di compensazione 3D con le coordinate macchina, inserire le coordinate di misura (se diverse dalle coordinate macchina) e le coordinate del pezzo (se differenti dalle coordinate macchina o la posizione del pezzo è cambiata).



Misura dell'errore volumetrico 3D col sistema Vettoriale

II. Informazioni da inserire da parte dell'utilizzatore

nella finestra del programma inserire i seguenti parametri come indicazione di riferimento:

Coordinate di Misura

X_m =

Y_m =

Z_m =

In partenza sono tutti a 0,

Coordinate del Pezzo:

X_p =

Y_p =

Z_p =

In partenza sono tutti a 0,

Lunghezza dei segmenti compensati:

D =

(Stesse unità del programma pezzo, in partenza=10)

Numero decimali =

in partenza sono 3 decimali

Cancel [], OK []

In partenza si assume che le coordinate macchina, le coordinate della misura e le coordinate del pezzo siano le stesse. Altrimenti, inserire le origini per le coordinate di misura e per il programma pezzo.

III. Interpolazione Lineare G01

Dopo aver letto G0 , i prossimi X, Y e Z sono le posizioni di partenza, X_o, Y_o e Z_o

Il prossimo G1 o G01, X_{Xe} Y_{Ye} Z_{Ze}, dove X_e, Y_e e Z_e sono le posizioni finali (e per ending). Nel caso di programma incrementale G91, sarà convertito in G90 programmazione assoluta.

Il percorso lineare da X_o, Y_o e Z_o a X_e, Y_e e Z_e sarà diviso in in numero di segmenti N così che la lunghezza di ogni segmento sia inferiore a D

Per esempio:

Programma pezzo originale:

Posizioni di partenza X_o Y_o Z_o

G1 X_{Xe} Y_{Ye} Z_{Ze}

Programma pezzo Segmentato:

Posizioni di partenza X_o Y_o Z_o

G1 X_{X1} Y_{Y1} Z_{Z1}

G1 X_{X2} Y_{Y2} Z_{Z2}

G1 X_{X3} Y_{Y3} Z_{Z3}

.....

.....

.....

G1 X_{XXN} Y_{YYN} Z_{ZZN},

Dove X_N=X_e, Y_N=Y_e, e Z_N=Z_e.

In modo simile, per il prossimo G1 o interpolazione lineare si usa la stessa formula. Se c'è una interpolazione circolare, G2, G3 prima di G1, viene usata la posizione finale come posizione iniziale del prossimo G1 o interpolazione lineare.

IV. Interpolazione Circolare G2, G3, G17, G18, G19

Dopo aver letto G2 o G3, il programma verifica G17, G18 e G19. G17 è il percorso circolare nel pianoXY, G18 è il percorso circolare nel pianoZX, G19 è il percorso circolare nel pianoYZ. Il seguente esempio è per il piano XY. In maniera simile si applica ai piani ZX e YZ. Dopo aver letto l'ultimo G1, G2 e G3, la posizione finale X,Y e Z è la posizione di partenza X_o,Y_o e Z_o per il prossimo G2 o G3 (G17 significa nel piano XY) X_{Xe} Y_{Ye} I_{Io} J_{Jo} dove X_e e Y_e sono le posizioni finali dopo il percorso circolare e I_o e J_o sono le distanze dal centro (X_c, Y_c) dell'arco circolare, dove X_c=X_o+I_o, Y_c=Y_o+J_o. Il percorso circolare viene diviso da X_o, e Y_o a X_e e Y_e in un numero di segmenti N, così che la lunghezza di ogni segmento è inferiore a D.

Programma pezzo originale:

Posizioni di partenza X_o Y_o

G3 X_{Xe} Y_{Ye} I_{Xc} J_{Yc}

Programma pezzo segmentato:

Posizioni di partenza X_o Y_o

G3 X_{XX1} Y_{YY1} I_{II1} J_{JJ1}

G3 X_{XX2} Y_{YY2} I_{II2} J_{JJ2}

G3 X_{XX3} Y_{YY3} I_{II3} J_{JJ3}

.....

.....

.....

G3 X_{XXN} Y_{YYN} I_{IIN} J_{JJN}

Dove X_N=X_e, e Y_N=Y_e,

In modo simile per G2 (direzione oraria), usa un incremento angolare negativo. Per G18 (piano ZX) sostituisce X con Z e Y con X, I con K e J con I. Per G 19 (piano YZ), sostituisce X con Y, Y con Z, I con J e J con K

Esempio :

Programma pezzo originale:

Posizioni di partenza $X_o = 4.000$ $Y_o = -2.000$

G17

G3 X-6.000 Y8.000 I-10.000 J0

Programma pezzo segmentato:

Posizioni di partenza $X_o = 4.000$ $Y_o = -2.000$

G3 X3.848 Y-0.264 I-10.000 J0.000

G3 X3.397 Y1.420 I-9.848 J-1.736

G3 X2.660 Y3.000 I-9.397 J-3.420

G3 X1.660 Y4.428 I-8.660 J-5.000

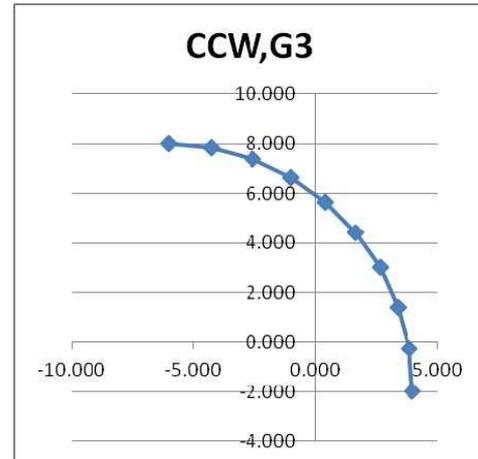
G3 X0.428 Y5.660 I-7.660 J-6.428

G3 X-1.000 Y6.660 I-6.428 J-7.660

G3 X-2.580 Y7.397 I-5.000 J-8.660

G3 X-4.264 Y7.848 I-3.420 J-9.397

G3 X-6.000 Y8.000 I-1.736 J-9.848



Programma pezzo originale:

Posizioni di partenza $X_o = 4.000$ $Y_o = -2.000$

G17

G2 X-6.000 Y-12.000 I-10.000 J0

Programma pezzo segmentato:

Posizioni di partenza $X_o = 4.000$ $Y_o = -2.000$

G2 X3.848 Y-3.736 I-10 J0

G2 X3.397 Y-5.420 I-9.848 J1.736

G2 X2.660 Y-7.000 I-9.396 J3.420

G2 X1.660 Y-8.428 I-8.660 J5

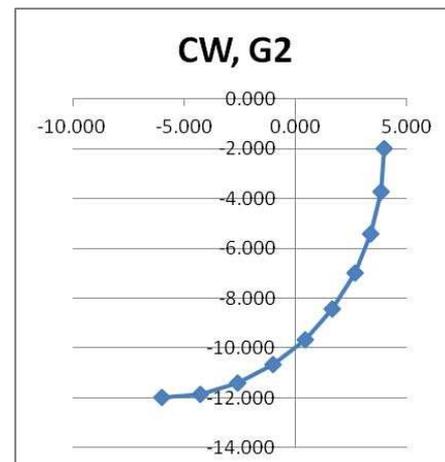
G2 X0.428 Y-9.660 I-7.660 J6.427

G2 X-1.000 Y-10.660 I-6.427 J7.660

G2 X-2.580 Y-11.397 I-5 J8.660

G2 X-4.264 Y-11.848 I-3.420 J9.396

G2 X-6.000 Y-12.000 I-1.736 J9.848



Dopo che è stato generato un programma pezzo segmentato, ogni coordinata X, Y, Z viene compensata usando la tabella di compensazione degli errori volumetrici 3D che è stata misurata in precedenza.

V. Generazione di un programma pezzo compensato

Prima di tutto misurare l'errore di posizionamento volumetrico 3D della macchina utensile da compensare per mezzo del metodo vettoriale. Poi elaborare col programma LDDM Analisi, i 4 file delle misure diagonali a passi sequenziali per calcolare gli errori di posizionamento volumetrico.

Le tabelle degli errori volumetrici 3D sono illustrati di seguito

ASSE X

Posizione	dxx	dxy	dxz
0.0	0.0	0.0	0.0
48.0	-0.0009479	-0.0015643	-0.0031717
96.0	-0.0038110	-0.0034922	-0.0040282
144.0	-0.0064295	-0.0060658	-0.0047497
192.0	-0.0074602	-0.0071584	-0.0059618
240.0	-0.0083099	-0.0088231	-0.0067169
288.0	-0.010453	-0.0098209	-0.0062623
336.0	-0.011624	-0.0082648	-0.0052214
384.0	-0.013692	-0.0061277	-0.0031075
432.0	-0.0144781	-0.003230	-0.0028492
480.0	-0.0142820	0.0	0.0

ASSE Y

Posizione	dyx	dyy	dyz
0.0	0.0	0.0	0.0
33.0	-0.0010738	-0.0039176	-0.0012255
66.0	0.0000804	-0.0042665	-0.0016438
99.0	0.0005462	-0.0072451	-0.0022323
132.0	0.0020470	-0.0088609	-0.0008964
165.0	0.0033270	-0.0094763	0.0000568
198.0	0.0043364	-0.0087056	-0.0002838
231.0	0.0066423	-0.0068262	0.0000806
264.0	0.0087687	-0.0071640	0.0006363
297.0	0.0100242	-0.0083214	0.0015772
330.0	0.0083370	-0.009819	0.0

ASSE Z

Posizione	dzx	dzy	dzz
0.0	0.0	0.0	0.0
45.0	0.0018057	0.0024344	0.0061428
90.0	0.0031612	0.0049691	0.0031962
135.0	0.004779	0.0078495	0.0017127
180.0	0.0061315	0.0107512	0.00146
225.0	0.0077631	0.0134967	0.0003928
270.0	0.0088113	0.0157202	-0.0035672
315.0	0.0106522	0.0181167	-0.0007415
360.0	0.0121948	0.0200268	-0.0070404
405.0	0.0140105	0.0220084	-0.0065292
450.0	0.0154910	0.023681	-0.01339

Nota1, significato dei valori:

Nell'esempio riportato se consideriamo una macchina con l'asse Z verticale avremo:

ASSE X: nella prima colonna la posizione nominale, nella seconda colonna dxx l'errore misurato lungo l'asse di movimento x (errore di posizionamento lineare), nella terza colonna dxy, l'errore lungo la direzione laterale (rettilinearità orizzontale) e nella quarta colonna dxz lo spostamento verso l'asse Z o rettilinearità verticale

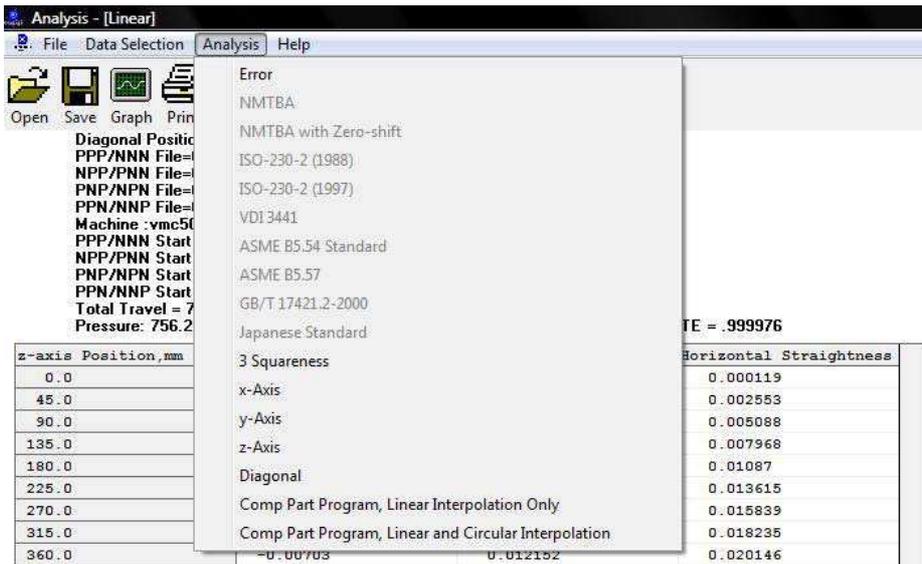
ASSE Y: l'errore di posizionamento lineare dyy è nella terza colonna mentre dyx e dyz sono rispettivamente la rettilinearità orizzontale e verticale.

ASSE Z : l'errore di posizionamento lineare è ora nell'ultima colonna dzz, mentre dzx e dzy sono le rettilinearità orizzontali

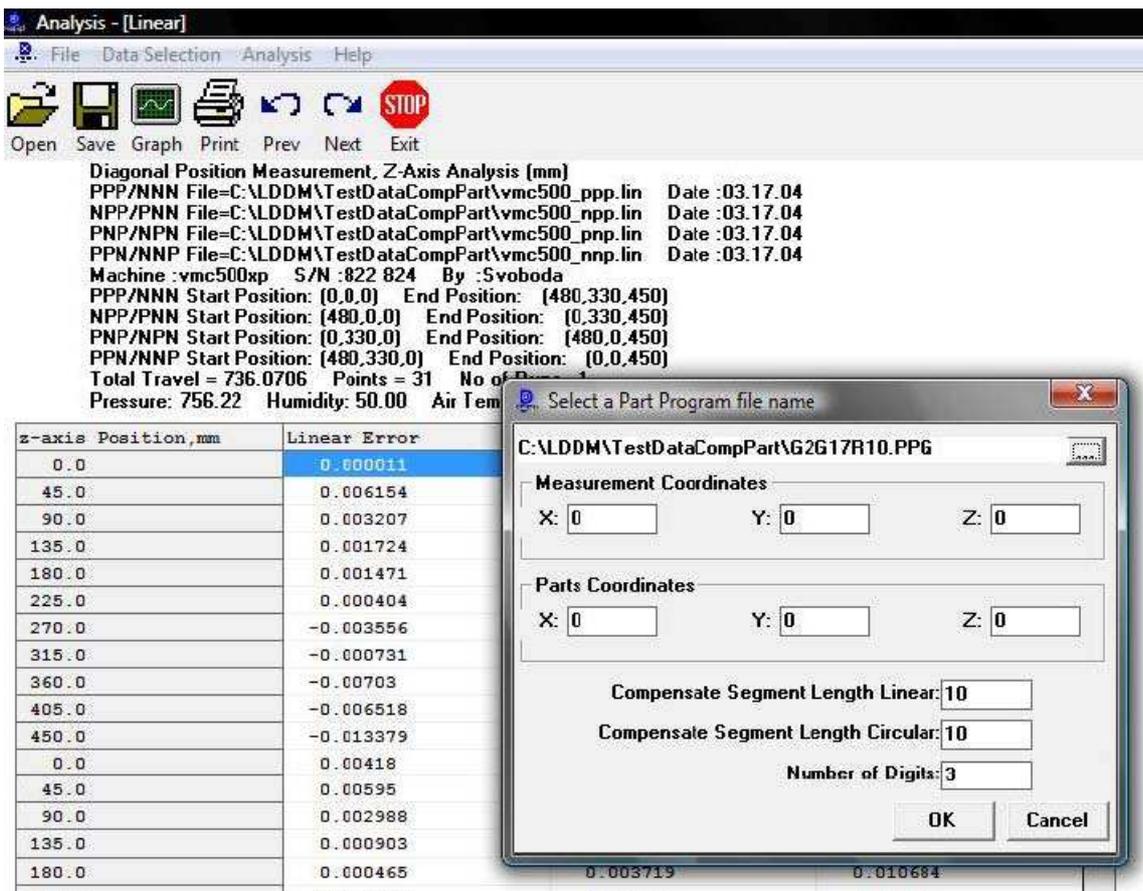
Nota2, rettilinearità e perpendicolarità:

Possiamo notare che le rettilinearità dxy, dxz e dyz partono da zero e ritornano a zero nell'ultimo punto, sono adattate; mentre le rettilinearità dyx, dzx e dzy non tornano a zero perchè contengono anche le informazioni di perpendicolarità.

Cliccare su “Analysis” and “Comp Part Program, Linear and Circular Interpolation” come illustrato di seguito.



Apparirà una finestra di dialogo per inserire il nome del programma pezzo da elaborare, le coordinate di inizio, la lunghezza dei segmenti e il numero dei decimali richiesti.



Cliccare su OK per generare un programma pezzo compensato, il file avrà lo stesso nome ma con estensione .VCM.

A sinistra, nelle tabelle il programma pezzo non compensato, a destra il programma compensato



Interpolazione lineare

Input G-code:	Output G-code:
% (LINEAR) G54 G90 G01 F3000 X0.Y0.Z0. M00 X100.Y70.Z80. Z240. X350. Y245. X450.Y315.Z360. Z0.Y0.X0. M30 %	% (LINEAR) G54 G90 G01 F3000 X0.Y0.Z0. M00 X100.001Y70.004Z80.002 Z240.007 X350.004 Y245. X449.993Y314.991Z360.008 Z0.Y0.X0. M30 %

Interpolazione circolare

Input G-code:	Output G-code:
(G2 Test Circular contour, R=10) G54 G90 G01 F3000 X4.000 Y-2.000 G17 G2 X-6.000 Y-12.000 I-10.000 J0 G1 X0 Y0 M30	(G2 Test Circular contour, R=10) G54 G90 G01 F3000 X4. Y-2. G17 G2 X3.848 Y-3.737 I-10. J0. G2 X3.397 Y-5.421 I-9.848 J1.736 G2 X2.66 Y-7.001 I-9.397 J3.42 G2 X1.66 Y-8.429 I-8.66 J5. G2 X0.428 Y-9.662 I-7.66 J6.428 G2 X-1. Y-10.662 I-6.428 J7.66 G2 X-2.58 Y-11.398 I-5. J8.66 G2 X-4.264 Y-11.85 I-3.42 J9.397 G2 X-6. Y-12.002 I-1.736 J9.848 G1 X-3. Y-6.001 G1 X0. Y0. M30