## Table of contents



# KinPolyTransP - Command options for kinematics - Path rounding with polynomial

# KinPolyTransP - Command options for kinematics - Path rounding with polynomial

## Basics

This command option switches an overblending where polynomial trajectories are inserted between successive travel commands permanently on for all travel commands until it is explicitly switched off.

Commanded motions are shortened and a polynomial motion (c2-continuous) is insulted in the resulting gap by ctrlX MOTION. Traveling takes place along this path (by complying with the programmed dynamic limits). A basic use case is the rounding of corners between two linear motions on a defined path (and thus with defined dynamic parameters). On the inserted, polynomial transition path, the same path dynamic limits apply as on the motion command before the corner.

Example:

If a motion 1 is commanded in X direction, following by a motion 2 in Y direction (i.e. before a corner with 90°), the function generates a c2-continuous path used to round the corner. On the rounded part, the same dynamic as for motion 1 can be used for traveling (as long as the axis limit values are not exceeded).

The calculation effort is considerably higher than for the command option blending ( ↘ “KinBlend, KinBlendP - Blending/permanent kinematic option”). Additionally, the traveled motion during blending ( ↘ “KinBlend, KinBlendP - Blending/permanent kinematic option”) is time-optimized (in contrast to PolyTrans).

In most cases, the “PolyTrans” function should be applied in combination with the command option “ContMotion” to generate a continuous velocity profile without decelerating to a standstill at command transitions.

## Parameter description

D1: Distance to the end position of the first motion command (corner E1/S2) at which blending can start.

D2: Distance after the start position of the second motion command (corner E1/S2) in which the motion has to be retraced to the path (end of blending).

Eps: Distance to the corner.

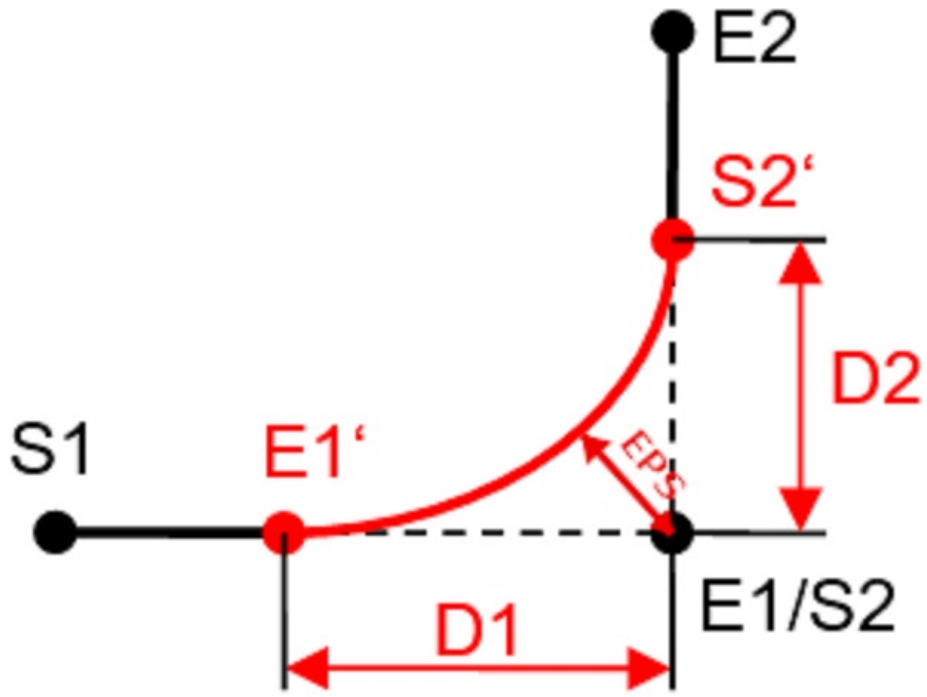


Fig. 71: Representation of the “PolyTrans” parameters

Both D1, D2 and Eps refer to the currently active FeedGroup (see ↘ “KinFeedGroupP - Specifying the feed group”). This means that only coordinates or axes that are part of the active FeedGroup are considered for the calculation of distances.

Set D1 or D2 to a very high value to blend the entire path (D1 and D2 are restricted to half of the path).

It is possible to either program D1 and D2 or Eps, bot not both. Do not set the used parameters to zero or omit them (depending on the used button). The used parameters have to be greater than zero.

If D1 and D2 are programmed, the originally programmed motions are shortened by the corresponding lengths (by max. half the length of the original path length) and the resulting gap is closed with a polynomial path (c2-continuous).

If Eps is programmed, the polynomial path is generated which does not deviate further than Eps from the originally programmed motions. The greatest deviation is the distance to the corner point. The distance can also be less due to geometric factors. The distance from start to end of the polynomial

path to the corner is also half the length of the original path length.

## Restrictions

If the coordinate system is changed between two motion commands (e.g. if a new PCS set is enabled), blending does not take place between these two motions. The corner is traveled completely. Even if one of the motions is a point-to-point motion (e.g. MoveDirect, MoveDirectAsync), blending is not used between these motions. Furthermore, blending does not take place if the output direction of the first motion is almost identical to the input direction of the second motion (aperture angle ≈ 180°) or if one of the motions is very short.

As a matter of principle, the command options “Blending” and “PolyTrans” rule each other out. An error is generated if both are simultaneously enabled for the same kinematics.

## PLC:

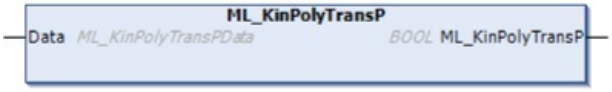


Fig. 72: ML_KinPolyTransP

STRUCT ML_KinPolyTransPData

| Name | Data type | Inherited from | Address | Initial value | Comment |
|---|---|---|---|---|---|
| In | ML_iKinPolyTransP | | | | Input: Command data of the command option |
| Out | ML_oCmdOptResult | | | | Output: Parameter of the command option |

STRUCT ML_iKinPolyTransP EXTENDS ML_iKinCmdBase

| Name | Data type | Inherited from | Address | Initial value | Comment |
|---|---|---|---|---|---|
| KinName | STRING(15) | ML_iKinCmdBase | | | Name of the kinematics used to execute the command option |
| Source | STRING(50) | ML_iKinCmdBase | | | Returns the command option source, e.g. PlcApplication1 is displayed in the diagnostic |

| | | | | |
|---|---|---|---|---|
| | | | | messages |
| SourceLine | ULINT | ML_iKinCmdBase | | Specifies the source line of the command; is displayed in the diagnostic messages |
| D1 | LREAL | | 0.0 | Distance to end position of the first motion command at which blending starts |
| D2 | LREAL | | 0.0 | Distance after the start position of the second motion command in which the motion has to be retraced to the path (end of blending). |
| Eps | LREAL | | 0.0 | Distance to the programmed boundary. |
| SwitchOff | BOOL | | FALSE | FALSE means that the blending option is enabled permanently, use TRUE to disable the command option again |

STRUCT ML_oCmdOptResult

| Name | Data type | Inherited from | Address | Initial value | Comment |
|---|---|---|---|---|---|
| Error | BOOL | | | | Error ID TRUE indicates that an error occurred |
| ErrorID | ULINT | | | | Specifies the type of occurred error: The upper 4 bits reflect the main diagnostics, the lower 4 bytes reflect the detailed diagnostics, 0 means no error |

## Data Layer:

Method: POST

URL: https://<ip-addr>/automation/api/v2/nodes/motion/kin/<kin_name>/cmd/opt-poly-trans

Payload:

```
{"type":"object","value":      {      "permType":"PermOn",      "dist1":1,      "dist2":1
}}
```

or

```
{"type":"object","value":      {      "permType":"PermOn",      "eps":1      }}
```

Only specify Dist1 and Dist2 or Eps during enabling.

## Python:

```
motion.kin_cmd_opt_poly_trans_p( kin=<kinName> [,d1=<D1>] [,d2=<D2>] [,eps=<Eps>])
```

Enables/disables the permanent command option for contour rounding by using polynomial paths to be inserted. The kinematics has to be assigned to the script instance (object has to be "attached"), see documentation Python Runtime App, chapter Python functions (https://docs.automation.boschrexroth.com/cdphelp?tkeyword=ctrlX_CORE_MANUAL_motion_python_additional)).

If none of the optional parameters is programmed for the path, the command option is disabled.

- <kinName> - string, kinematic object name
- <D1> - double; max. distance before the corner for rounding
- <D2> - double; max. distance behind the corner for rounding
- <Eps> - double; max. distance of the generated polynomial path from the corner

To enable the command option, program either D1 and D2 with a value greater than 0 or Eps.

## BundleIF:

```
//! Create a command option for inserting polynomial commands between successively move co
mmands (KIN_CMD_OPT_POLY_TRANS)
//!
//! @param[in]  sourceInfo      Command source information
//! @param[in]  duration        Do this command option once, permanent or remove it?
//! @param[in]  kinName         Name of the motion object
//! @param[in]  preCornerDist   Distance to shorten the end of the precorner (D1)
//! @param[in]  postCornerDist  Distance to shorten the start of the postcorner (D2)
//! @param[in]  eps             DEPRECATED (can be set to PolyTransMode::MODE_L1L2 always
//! @param[in]  mode            Mode L1L2 or EPS
//! @return                     Was everything okay?
virtual MotionResult kinPolyTrans(const dia::Cmd-SourceInfo& sourceInfo, cmd::CmdOptPermTy
pe duration, const char* kinName, double preCornerDist, double postCornerDist, double eps,
cmd::PolyTransMode mode) const = 0;
```