FICO^{TM} Xpress
Optimization Suite

Xpress Knitro Controls Reference Manual

Manual

Release version 29.01

Last update 14 February, 2016

www.fico.com | Make every decision count^{TM}

Xpress-Knitro

Deliverable Version: A

Last Revised: 14 February, 2016

Version version 29.01

Contents

CHAPTER 1

Xpress Knitro Control Parameters

This manual provides a full list of the controls accepted by Xpress for setting Knitro parameters. KNITRO has a great number and variety of user option settings and although it tries to choose the best settings by default, often significant performance improvements can be realized by choosing some non-default option settings.

XKTR_PARAM_ALGORITHM Indicates which algorithm to use to solve the problem p. 8

XKTR_PARAM_BAR_DIRECTINTERVAL Controls the maximum number of consecutive conjugate gradient (CG) steps before KNITRO will try to enforce that a step is taken using direct linear algebra. p. 8

XKTR_PARAM_BAR_FEASIBLE Specifies whether special emphasis is placed on getting and staying feasible in the interior-point algorithms. p. 8

XKTR_PARAM_BAR_FEASMODETOL Specifies the tolerance in equation that determines whether KNITRO will force subsequent iterates to remain feasible. p. 4

XKTR_PARAM_BAR_INITMU Specifies the initial value for the barrier parameter : used with the barrier algorithms. This option has no effect on the Active Set algorithm. p. 4

XKTR_PARAM_BAR_INITPT Indicates whether an initial point strategy is used with barrier algorithms. p. 9

XKTR_PARAM_BAR_MAXBACKTRACK Indicates the maximum allowable number of backtracks during the linesearch of the Interior/Direct algorithm before reverting to a CG step. p. 9

XKTR_PARAM_BAR_MAXCROSSIT Specifies the maximum number of crossover iterations before termination. p. 9

XKTR_PARAM_BAR_MAXREFACTOR Indicates the maximum number of refactorizations of the KKT system per iteration of the Interior/Direct algorithm before reverting to a CG step. p. 10

XKTR_PARAM_BAR_MURULE Indicates which strategy to use for modifying the barrier parameter mu in the barrier algorithms. p. 10

XKTR_PARAM_BAR_PENCONS Indicates whether a penalty approach is applied to the constraints. p. 11

XKTR_PARAM_BAR_PENRULE Indicates which penalty parameter strategy to use for determining whether or not to accept a trial iterate. p. 11

XKTR_PARAM_BAR_SWITCHRULE Indicates whether or not the barrier algorithms will allow switching from an optimality phase to a pure feasibility phase. p. 12

XKTR_PARAM_MIP_MAXNODES Specifies the maximum number of nodes explored. p. 16

XKTR_PARAM_MIP_MAXSOLVES Specifies the maximum number of subproblem solves allowed (0 means no limit). p. 16

XKTR_PARAM_MIP_METHOD Specifies which MIP method to use. p. 17

XKTR_PARAM_MIP_OUTINTERVAL Specifies node printing interval for XKTR_PARAM_MIP_OUTLEVEL when XKTR_PARAM_MIP_OUTLEVEL > 0. p. 17

XKTR_PARAM_MIP_OUTLEVEL Specifies how much MIP information to print. p. 17

XKTR_PARAM_MIP_PSEUDOINIT Specifies the method used to initialize pseudo-costs corresponding to variables that have not yet been branched on in the MIP method. p. 17

XKTR_PARAM_MIP_ROOTALG Specifies which algorithm to use for the root node solve in MIP (same options as XKTR_PARAM_ALGORITHM user option). p. 18

XKTR_PARAM_MIP_ROUNDING Specifies the MIP rounding rule to apply. p. 18

XKTR_PARAM_MIP_SELECTRULE Specifies the MIP select rule for choosing the next node in the branch and bound tree. p. 18

XKTR_PARAM_MIP_STRONG_CANDLIM Specifies the maximum number of candidates to explore for MIP strong branching. p. 19

XKTR_PARAM_MIP_STRONG_LEVEL Specifies the maximum number of tree levels on which to perform MIP strong branching. p. 19

XKTR_PARAM_MIP_STRONG_MAXIT Specifies the maximum number of iterations to allow for MIP strong branching solves. p. 19

XKTR_PARAM_MIP_TERMINATE Specifies conditions for terminating the MIP algorithm. p. 19

XKTR_PARAM_OBJRANGE Specifies the extreme limits of the objective function for purposes of determining unboundedness. p. 6

XKTR_PARAM_OPTTOL Specifies the final relative stopping tolerance for the KKT (optimality) error. p. 6

XKTR_PARAM_OPTTOLABS Specifies the final absolute stopping tolerance for the KKT (optimality) error. p. 6

XKTR_PARAM_OUTLEV Controls the level of output produced by KNITRO. p. 19

XKTR_PARAM_PRESOLVE Determine whether or not to use the KNITRO presolver to try to simplify the model by removing variables or constraints. Specifies conditions for terminating the MIP algorithm. p. 20

XKTR_PARAM_PRESOLVE_TOL Determines the tolerance used by the KNITRO presolver to remove variables and constraints from the model. p. 7

XKTR_PARAM_SCALE Performs a scaling of the objective and constraint functions based on their values at the initial point. p. 20

XKTR_PARAM_SOC Specifies whether or not to try second order corrections (SOC). p. 20

XKTR_PARAM_XTOL The optimization process will terminate if the relative change in all components of the solution point estimate is less than xtol. p. 7

1.1 Double control parameters

These double control parameters can be set using XSLPsetdblcontrol using the Xpress-SLP API, XNLPsetsolverdoublecontrol using the XNLP API and setparam in Mosel using module mmxnlp.

XKTR_PARAM_BAR_FEASMODETOL

Description Specifies the tolerance in equation that determines whether KNITRO will force subsequent iterates to remain feasible.

Type Double

Note The tolerance applies to all inequality constraints in the problem. This option only has an effect if option XKTR_PARAM_BAR_FEASIBLE = stay or XKTR_PARAM_BAR_FEASIBLE = get_stay.

Default value 1.0e-4

XKTR_PARAM_BAR_INITMU

Description Specifies the initial value for the barrier parameter : μ used with the barrier algorithms. This option has no effect on the Active Set algorithm.

Type Double

Default value 1.0e-1

XKTR_PARAM_DELTA

Description Specifies the initial trust region radius scaling factor used to determine the initial trust region size.

Type Double

Default value 1.0e0

XKTR_PARAM_DELTA

Description Specifies the initial trust region radius scaling factor used to determine the initial trust region size.

Type Double

Default value 1.0e0

XKTR_PARAM_FEASTOL

Description	Specifies the final relative stopping tolerance for the feasibility error.
Type	Double
Note	Smaller values of feastol result in a higher degree of accuracy in the solution with respect to feasibility.
Default value	1.0e-6

XKTR_PARAM_FEASTOLABS

Description	Specifies the final absolute stopping tolerance for the feasibility error.
Type	Double
Note	Smaller values of feastol_abs result in a higher degree of accuracy in the solution with respect to feasibility.
Default value	0.0e0

XKTR_PARAM_INFEASTOL

Description	Specifies the (relative) tolerance used for declaring infeasibility of a model.
Type	Double
Note	Smaller values of infeastol make it more difficult to satisfy the conditions KNITRO uses for detecting infeasible models. If you believe KNITRO incorrectly declares a model to be infeasible, then you should try a smaller value for infeastol.
Default value	1.0e-8

XKTR_PARAM_MIP_INTEGERTOL

Description	This value specifies the threshold for deciding whether or not a variable is determined to be an integer.
Type	Double
Default value	1.0e-8

XKTR_PARAM_MIP_INTGAPABS

Description The absolute integrality gap stop tolerance for MIP.

Type Double

Default value 1.0e-6

XKTR_PARAM_MIP_INTGAPREL

Description The relative integrality gap stop tolerance for MIP.

Type Double

Default value 1.0e-6

XKTR_PARAM_OBJRANGE

Description Specifies the extreme limits of the objective function for purposes of determining unboundedness.

Type Double

Note If the magnitude of the objective function becomes greater than objrange for a feasible iterate, then the problem is determined to be unbounded and KNITRO proceeds no further.

Default value 1.0e20

XKTR_PARAM_OPTTOL

Description Specifies the final relative stopping tolerance for the KKT (optimality) error.

Type Double

Note Smaller values of opttol result in a higher degree of accuracy in the solution with respect to optimality.

Default value 1.0e-6

XKTR_PARAM_OPTTOLABS

Description Specifies the final absolute stopping tolerance for the KKT (optimality) error.

Type Double

Note Smaller values of opttol_abs result in a higher degree of accuracy in the solution with respect to optimality.

Default value 0.0e0

XKTR_PARAM_PRESOLVE_TOL

Description Determines the tolerance used by the KNITRO presolver to remove variables and constraints from the model.

Type Double

Note If you believe the KNITRO presolver is incorrectly modifying the model, use a smaller value for this tolerance (or turn the presolver off).

Default value 1.0e-6

XKTR_PARAM_XTOL

Description The optimization process will terminate if the relative change in all components of the solution point estimate is less than xtol.

Type Double

Note If using the Interior/Direct or Interior/CG algorithm and the barrier parameter is still large, KNITRO will first try decreasing the barrier parameter before terminating.

Default value 1.0e-15

1.2 Integer control parameters

These integer control parameters can be set using XSLPsetintcontrol using the Xpress-SLP API, XNLPsetsolverintcontrol using the XNLP API and setparam in Mosel using module mmxnlp.

XKTR_PARAM_ALGORITHM

Description Indicates which algorithm to use to solve the problem

Type Integer

Values

`0`	(auto) let KNITRO automatically choose an algorithm, based on the problem characteristics.
`1`	(direct) use the Interior/Direct algorithm.
`2`	(cg) use the Interior/CG algorithm.
`3`	(active) use the Active Set algorithm.
`4`	(sqp) use the SQP algorithm.
`5`	(multi) run all algorithms, perhaps in parallel.

Default value 0

XKTR_PARAM_BAR_DIRECTINTERVAL

Description Controls the maximum number of consecutive conjugate gradient (CG) steps before KNITRO will try to enforce that a step is taken using direct linear algebra.

Type Integer

Note This option is only valid for the Interior/Direct algorithm and may be useful on problems where KNITRO appears to be taking lots of conjugate gradient steps. Setting bar_directinterval to 0 will try to enforce that only direct steps are taken which may produce better results on some problems.

Default value 10

XKTR_PARAM_BAR_FEASIBLE

Description Specifies whether special emphasis is placed on getting and staying feasible in the interior-point algorithms.

Type Integer

Values

`0`	(no) No special emphasis on feasibility.
`1`	(stay) Iterates must satisfy inequality constraints once they become sufficiently feasible.
`2`	(get) Special emphasis is placed on getting feasible before trying to optimize.
`3`	(get_stay) Implement both options 1 and 2 above.

Note This option can only be used with the Interior/Direct and Interior/CG algorithms. If bar_feasible = stay or bar_feasible = get_stay, this will activate the feasible version of KNITRO. The feasible version of KNITRO will force iterates to strictly satisfy inequalities, but does not require satisfaction of equality constraints at intermediate iterates. This option and the honorbnds option may be useful in applications where functions are undefined outside the region defined by inequalities. The initial point must satisfy inequalities to a sufficient degree; if not, KNITRO may generate infeasible iterates and does not switch to the feasible version until a sufficiently feasible point is found. Sufficient satisfaction occurs at a point x if it is true for all inequalities that cl + tol $\leq$ c(x) $\leq$ cu - tol The constant tol is determined by the option bar_feasmodetol. If bar_feasible = get or bar_feasible = get_stay, KNITRO will place special emphasis on first trying to get feasible before trying to optimize.

Default value 0

XKTR_PARAM_BAR_INITPT

Description Indicates whether an initial point strategy is used with barrier algorithms.

Type Integer

Values

0	(auto) Let KNITRO automatically choose the strategy.
1	(yes) Shift the initial slacks and multipliers to improve barrier algorithm performance.
2	(no) Do no alter the initial slacks and multipliers.

Note This option has no effect on the Active Set algorithm.

Default value 0

XKTR_PARAM_BAR_MAXBACKTRACK

Description Indicates the maximum allowable number of backtracks during the linesearch of the Interior/Direct algorithm before reverting to a CG step.

Type Integer

Note Increasing this value will make the Interior/Direct algorithm less likely to take CG steps. If the Interior/Direct algorithm is taking a large number of CG steps (as indicated by a positive value for 'Gits' in the output), this may improve performance. This option has no effect on the Active Set algorithm.

Default value 3

XKTR_PARAM_BAR_MAXCROSSIT

Description Specifies the maximum number of crossover iterations before termination.

Type Integer

Note If the value is positive and the algorithm in operation is Interior/Direct or Interior/CG, then KNITRO will crossover to the Active Set algorithm near the solution. The Active Set algorithm will then perform at most bar_maxcrossit iterations to get a more exact solution. If the value is 0, no Active Set crossover occurs and the interior-point solution is the final result. If Active Set crossover is unable to improve the approximate interior-point solution, then KNITRO will restore the interior-point solution. In some cases (especially on large-scale problems or difficult degenerate problems) the cost of the crossover procedure may be significant - for this reason, crossover is disabled by default. Enabling crossover generally provides a more accurate solution than Interior/Direct or Interior/CG.

Default value 0

XKTR_PARAM_BAR_MAXREFACTOR

Description Indicates the maximum number of refactorizations of the KKT system per iteration of the Interior/Direct algorithm before reverting to a CG step.

Type Integer

Note These refactorizations are performed if negative curvature is detected in the model. Rather than reverting to a CG step, the Hessian matrix is modified in an attempt to make the subproblem convex and then the KKT system is refactorized. Increasing this value will make the Interior/Direct algorithm less likely to take CG steps. If the Interior/Direct algorithm is taking a large number of CG steps (as indicated by a positive value for "CGits" in the output), this may improve performance. This option has no effect on the Active Set algorithm.

Default value -1

XKTR_PARAM_BAR_MURULE

Description Indicates which strategy to use for modifying the barrier parameter mu in the barrier algorithms.

Type Integer

Values

0 (auto) Let KNITRO automatically choose the strategy.

1 (monotone) Monotonically decrease the barrier parameter. Available for both barrier algorithms.

2 (adaptive) Use an adaptive rule based on the complementarity gap to determine the value of the barrier parameter. Available for both barrier algorithms.

3 (probing) Use a probing (affine-scaling) step to dynamically determine the barrier parameter. Available only for the Interior/Direct algorithm.

4 (dampmpc) Use a Mehrotra predictor-corrector type rule to determine the barrier parameter, with safeguards on the corrector step. Available only for the Interior/Direct algorithm.

5 (fullmpc) Use a Mehrotra predictor-corrector type rule to determine the barrier parameter, without safeguards on the corrector step. Available only for the Interior/Direct algorithm.

6 (quality) Minimize a quality function at each iteration to determine the barrier parameter. Available only for the Interior/Direct algorithm.

Note Not all strategies are available for both barrier algorithms. This option has no effect on the Active Set algorithm.

Default value 0

XKTR_PARAM_BAR_PENCONS

Description Indicates whether a penalty approach is applied to the constraints.

Type Integer

Values

0 (auto) Let KNITRO automatically choose the strategy.

1 (none) No constraints are penalized.

2 (all) A penalty approach is applied to all general constraints.

Note Using a penalty approach may be helpful when the problem has degenerate or difficult constraints. It may also help to more quickly identify infeasible problems, or achieve feasibility in problems with difficult constraints. This option has no effect on the Active Set algorithm.

Default value 0

XKTR_PARAM_BAR_PENRULE

Description Indicates which penalty parameter strategy to use for determining whether or not to accept a trial iterate.

Type Integer

Values

0 (auto) Let KNITRO automatically choose the strategy.

1 (single) Use a single penalty parameter in the merit function to weight feasibility versus optimality.

2 (flex) Use a more tolerant and flexible step acceptance procedure based on a range of penalty parameter values.

Note This option has no effect on the Active Set algorithm.

Default value 0

XKTR_PARAM_BAR_SWITCHRULE

Description Indicates whether or not the barrier algorithms will allow switching from an optimality phase to a pure feasibility phase.

Type Integer

Values

0 (auto) Let KNITRO determine the switching procedure.
1 (never) Never switch to feasibility phase.
2 (level1) Allow switches to feasibility phase.
3 (level2) Use a more aggressive switching rule.

Note This option has no effect on the Active Set algorithm.

Default value 0

XKTR_PARAM_GRADOPT

Description Specifies how to compute the gradients of the objective and constraint functions.

Type Integer

Values

1 (exact) User provides a routine for computing the exact gradients.
2 (forward) KNITRO computes gradients by forward finite-differences.
3 (central) KNITRO computes gradients by central finite differences.

Note It is highly recommended to provide exact gradients if at all possible as this greatly impacts the performance of the code.

Default value 1

XKTR_PARAM_HESSOPT

Description Specifies how to compute the (approximate) Hessian of the Lagrangian.

Type Integer

Values

1 (exact) User provides a routine for computing the exact Hessian.
2 (bfgs) KNITRO computes a (dense) quasi-Newton BFGS Hessian.
3 (sr1) KNITRO computes a (dense) quasi-Newton SR1 Hessian.
4 (finite_diff) KNITRO computes Hessian-vector products using finite-differences.
5 (product) User provides a routine to compute the Hessian-vector products.
6 (lbfgs) KNITRO computes a limited-memory quasi-Newton BFGS Hessian (its size is determined by the option lmsize).

Note Options hessopt = 4 and hessopt = 5 are not available with the Interior/Direct algorithm. KNITRO usually performs best when the user provides exact Hessians (hessopt = 1) or exact Hessian-vector products (hessopt = 5). If neither can be provided but exact gradients are available (i.e., gradopt = 1), then hessopt = 4 is recommended. This option is comparable in terms of robustness to the exact Hessian option and typically not much slower in terms of time, provided that gradient evaluations are not a dominant cost. If exact gradients cannot be provided, then one of the quasi-Newton options is preferred. Options hessopt = 2 and hessopt = 3 are only recommended for small problems ($n \leq 1000$) since they require working with a dense Hessian approximation. Option hessopt = 6 should be used for large problems.

Default value 1

XKTR_PARAM_HONORBNDS

Description Indicates whether or not to enforce satisfaction of simple variable bounds throughout the optimization.

Type Integer

Values

0	(no) KNITRO does not require that the bounds on the variables be satisfied at intermediate iterates.
1	(always) KNITRO enforces that the initial point and all subsequent solution estimates satisfy the bounds on the variables.
2	(initpt) KNITRO enforces that the initial point satisfies the bounds on the variables.

Note This option and the bar_feasible option may be useful in applications where functions are undefined outside the region defined by inequalities.

Default value 2

XKTR_PARAM_LMSIZE

Description Specifies the number of limited memory pairs stored when approximating the Hessian using the limited-memory quasi-Newton BFGS option.

Type Integer

Note The value must be between 1 and 100 and is only used with XKTR_PARAM_HESSOPT = 6. Larger values may give a more accurate, but more expensive, Hessian approximation. Smaller values may give a less accurate, but faster, Hessian approximation. When using the limited memory BFGS approach it is recommended to experiment with different values of this parameter.

Default value 10

XKTR_PARAM_MAXCGIT

Description	Specifies the number of limited memory pairs stored when approximating the Hessian using the limited-memory quasi-Newton BFGS option.	
Type	Integer	
Values	`0`	Let KNITRO automatically choose a value based on the problem size.
	`n`	At most n>0 CG iterations may be performed during one minor iteration of KNITRO.
Default value	0	

XKTR_PARAM_MAXIT

Description	Specifies the maximum number of iterations before termination.	
Type	Integer	
Values	`0`	Let KNITRO automatically choose a value based on the problem type. Currently KNITRO sets this value to 10000 for LPs/NLPs and 3000 for MIP problems.
	`n`	At most n>0 iterations may be performed before terminating.
Default value	0	

XKTR_PARAM_MIP_BRANCHRULE

Description	Specifies which branching rule to use for MIP branch and bound procedure.	
Type	Integer	
Values	`0`	(auto) Let KNITRO automatically choose the branching rule.
	`1`	(most_frac) Use most fractional (most infeasible) branching.
	`2`	(pseudcost) Use pseudo-cost branching.
	`3`	(strong) Use strong branching (see options `XKTR_PARAM_MIP_STRONG_CANDLIM`, `XKTR_PARAM_MIP_STRONG_LEVEL`, `XKTR_PARAM_MIP_STRONG_MAXIT` for further control of strong branching procedure).
Default value	0	

XKTR_PARAM_MIP_GUB_BRANCH

Description	Specifies whether or not to branch on generalized upper bounds (GUBs).
Type	Integer

Values

0	(no) Do not branch on GUBs.
1	(yes) Allow branching on GUBs.

Default value 0

XKTR_PARAM_MIP_HEURISTIC

Description Specifies which MIP heuristic search approach to apply to try to find an initial integer feasible point.

Type Integer

Values

0	(auto) Let KNITRO choose the heuristic to apply (if any).
1	(none) No heuristic search applied.
2	(feaspump) Apply feasibility pump heuristic.
3	(mpec) Apply heuristic based on MPEC formulation.

Note If a heuristic search procedure is enabled, it will run for at most mip_heuristic_maxit iterations, before starting the branch and bound procedure.

Default value 0

XKTR_PARAM_MIP_HEURISTIC_MAXIT

Description Specifies the maximum number of iterations to allow for MIP heuristic, if one is enabled.

Type Integer

Default value 100

XKTR_PARAM_MIP_IMPLICATNS

Description Specifies whether or not to add constraints to the MIP derived from logical implications.

Type Integer

Values

0	(no) Do not add constraints from logical implications.
1	(yes) KNITRO adds constraints from logical implications.

Default value 1

XKTR_PARAM_MIP_KNAPSACK

Description Specifies rules for adding MIP knapsack cuts.

Type Integer

Values

`0`	(none) Do not add knapsack cuts.
`1`	(ineqs) Add cuts derived from inequalities only.
`2`	(ineqs_eqs) Add cuts derived from both inequalities and equalities.

Default value 1

XKTR_PARAM_MIP_LPALG

Description Specifies which algorithm to use for any linear programming (LP) subproblem solves that may occur in the MIP branch and bound procedure.

Type Integer

Values

`0`	(auto) Let KNITRO automatically choose an algorithm, based on the problem characteristics.
`1`	(direct) Use the Interior/Direct (barrier) algorithm.
`2`	(cg) Use the Interior/CG (barrier) algorithm.
`3`	(active) Use the Active Set (simplex) algorithm.

Note LP subproblems may arise if the problem is a mixed integer linear program (MILP), or if using `XKTR_PARAM_MIP_METHOD` = HQG. (Nonlinear programming subproblems use the algorithm specified by the algorithm option.)

Default value 0

XKTR_PARAM_MIP_MAXNODES

Description Specifies the maximum number of nodes explored.

Type Integer

Note Zero vealue means no limit.

Default value 100000

XKTR_PARAM_MIP_MAXSOLVES

Description Specifies the maximum number of subproblem solves allowed (0 means no limit).

Type Integer

Default value 200000

XKTR_PARAM_MIP_METHOD

Description Specifies which MIP method to use.

Type Integer

Values
0 (auto) Let KNITRO automatically choose the method.
1 (BB) Use the standard branch and bound method.
2 (HQG) Use the hybrid Quesada-Grossman method (for convex, nonlinear problems only).

Default value 0

XKTR_PARAM_MIP_OUTINTERVAL

Description Specifies node printing interval for XKTR_PARAM_MIP_OUTLEVEL when XKTR_PARAM_MIP_OUTLEVEL > 0.

Type Integer

Values
0 Print output every node.
2 Print output every 2nd node.
N Print output every Nth node.

Default value 10

XKTR_PARAM_MIP_OUTLEVEL

Description Specifies how much MIP information to print.

Type Integer

Values
0 (none) Do not print any MIP node information.
1 (iters) Print one line of output for every node.

Default value 1

XKTR_PARAM_MIP_PSEUDOINIT

Description Specifies the method used to initialize pseudo-costs corresponding to variables that have not yet been branched on in the MIP method.

Type Integer

Values

0	Let KNITRO automatically choose the method.
1	Initialize using the average value of computed pseudo-costs.
2	Initialize using strong branching.

Default value 0

XKTR_PARAM_MIP_ROOTALG

Description Specifies which algorithm to use for the root node solve in MIP (same options as `XKTR_PARAM_ALGORITHM` user option).

Type Integer

Default value 0

XKTR_PARAM_MIP_ROUNDING

Description Specifies the MIP rounding rule to apply.

Type Integer

Values

0	(auto) Let KNITRO choose the rounding rule.
1	(none) Do not round if a node is infeasible.
2	(heur_only) Round using a fast heuristic only.
3	(nlp_sometimes) Round and solve a subproblem if likely to succeed.
4	(nlp_always) Always round and solve a subproblem.

Default value 0

XKTR_PARAM_MIP_SELECTRULE

Description Specifies the MIP select rule for choosing the next node in the branch and bound tree.

Type Integer

Values

0	(auto) Let KNITRO choose the node selection rule.
1	(depth_first) Search the tree using a depth first procedure.
2	(best_bound) Select the node with the best relaxation bound.
3	(combo_1) Use depth first unless pruned, then best bound.

Default value 0

XKTR_PARAM_MIP_STRONG_CANDLIM

Description Specifies the maximum number of candidates to explore for MIP strong branching.

Type Integer

Default value 10

XKTR_PARAM_MIP_STRONG_LEVEL

Description Specifies the maximum number of tree levels on which to perform MIP strong branching.

Type Integer

Default value 10

XKTR_PARAM_MIP_STRONG_MAXIT

Description Specifies the maximum number of iterations to allow for MIP strong branching solves.

Type Integer

Default value 1000

XKTR_PARAM_MIP_TERMINATE

Description Specifies conditions for terminating the MIP algorithm.

Type Integer

Values

`0` (optimal) Terminate at optimum.

`1` (feasible) Terminate at first integer feasible point.

Default value 0

XKTR_PARAM_OUTLEV

Description Controls the level of output produced by KNITRO.

Type Integer

Values
0 (none) Printing of all output is suppressed.
1 (summary) Print only summary information.
2 (iter_10) Print basic information every 10 iterations.
3 (iter) Print basic information at each iteration.
4 (iter_verbose) Print basic information and the function count at each iteration.
5 (iter_x) Print all the above, and the values of the solution vector x.
6 (all) Print all the above, and the values of the constraints c at x and the Lagrange multipliers lambda.

Default value 2

XKTR_PARAM_PRESOLVE

Description Determine whether or not to use the KNITRO presolver to try to simplify the model by removing variables or constraints. Specifies conditions for terminating the MIP algorithm.

Type Integer

Values
0 (none) Do not use KNITRO presolver.
1 (basic) Use the KNITRO basic presolver.

Default value 1

XKTR_PARAM_SCALE

Description Performs a scaling of the objective and constraint functions based on their values at the initial point.

Type Integer

Values
0 (no) No scaling is performed.
1 (yes) KNITRO is allowed to scale the objective function and constraints.

Note If scaling is performed, all internal computations, including the stopping tests, are based on the scaled values.

Default value 1

XKTR_PARAM_SOC

Description Specifies whether or not to try second order corrections (SOC).

Type Integer

Values	0	(no) No second order correction steps are attempted.
	1	(maybe) Second order correction steps may be attempted on some iterations.
	2	(yes) Second order correction steps are always attempted if the original step is rejected and there are nonlinear constraints.

Note A second order correction may be beneficial for problems with highly nonlinear constraints.

Default value 1

Appendix A

Contacting FICO

FICO provides clients with support and services for all our products. Refer to the following sections for more information.

Product support

FICO offers technical support and services ranging from self-help tools to direct assistance with a FICO technical support engineer. Support is available to all clients who have purchased a FICO product and have an active support or maintenance contract. You can find support contact information on the Product Support home page (www.fico.com/support).

On the Product Support home page, you can also register for credentials to log on to FICO Online Support, our web-based support tool to access Product Support 24x7 from anywhere in the world. Using FICO Online Support, you can enter cases online, track them through resolution, find articles in the FICO Knowledge Base, and query known issues.

Please include *'Xpress'* in the subject line of your support queries.

Product education

FICO Product Education is the principal provider of product training for our clients and partners. Product Education offers instructor-led classroom courses, web-based training, seminars, and training tools for both new user enablement and ongoing performance support. For additional information, visit the Product Education homepage at www.fico.com/en/product-training or email producteducation@fico.com.

Product documentation

FICO continually looks for new ways to improve and enhance the value of the products and services we provide. If you have comments or suggestions regarding how we can improve this documentation, let us know by sending your suggestions to techpubs@fico.com.

Sales and maintenance

USA, CANADA AND ALL AMERICAS

Email: XpressSalesUS@fico.com

WORLDWIDE

Email: XpressSalesUK@fico.com

Tel: +44 207 940 8718
Fax: +44 870 420 3601

Xpress Optimization, FICO
FICO House
International Square
Starley Way
Birmingham B37 7GN
UK

Related services

Strategy Consulting: Included in your contract with FICO may be a specified amount of consulting time to assist you in using FICO Optimization Modeler to meet your business needs. Additional consulting time can be arranged by contract.

Conferences and Seminars: FICO offers conferences and seminars on our products and services. For announcements concerning these events, go to www.fico.com or contact your FICO account representative.

About FICO

FICO (NYSE:FICO) delivers superior predictive analytics solutions that drive smarter decisions. The company's groundbreaking use of mathematics to predict consumer behavior has transformed entire industries and revolutionized the way risk is managed and products are marketed. FICO's innovative solutions include the FICO® Score—the standard measure of consumer credit risk in the United States—along with industry-leading solutions for managing credit accounts, identifying and minimizing the impact of fraud, and customizing consumer offers with pinpoint accuracy. Most of the world's top banks, as well as leading insurers, retailers, pharmaceutical companies, and government agencies, rely on FICO solutions to accelerate growth, control risk, boost profits, and meet regulatory and competitive demands. FICO also helps millions of individuals manage their personal credit health through www.myfico.com. Learn more at www.fico.com. FICO: Make every decision count^{TM}.

Index