HELI-X

User's Guide

Version 10.1 (2023-12-14)

Licence

HELI-X R/C Flight Simulator. Copyright ï¿½ 2008-2023 by www.heli-x.net (info@heli-x.net). All rights reserved.

This version of HELI-X is not freeware. You can download the demo version of HELI-X from www.heli-x.net and the sites which are linked from this site for testing purposes. The full version of HELI-X can be obtained via www.heli-x.net. For both versions, you only obtain the right to use HELI-X privately on your computers.

It is NOT allowed to make HELI-X available for download or to distribute or sell it. You may not rent, lease, modify, translate, reverse engineer, decompile, disassemble, or create derivative works based on HELI-X.

The usage of HELI-X without registration at www.heli-x.net is not legal. If you are not sure whether you are allowed to use your copy of HELI-X, ask at info@heli-x.net.

If you are interested in a commercial use of HELI-X, please contact info@heli-x.net.

This program was built using Java, jMonkeyEngine, Jogl, Joal, LWJGL, Jdom, the java networking library, Jama, Jagatoo, Gluegen, jSerialComm and JInput. Many thanks to all developers involved in these great projects. All licenses for these projects are stored in the particular folder.

Warranty Disclaimers and Liability Limitations

HELI-X, and any and all accompanying software, files, data and materials, are distributed and provided "AS IS" and with no warranties of any kind, whether express or implied. You acknowledge that good data processing procedure dictates that any program, including HELI-X, must be thoroughly tested with non-critical data before you rely on it, and you hereby assume the entire risk of using the program. The same is true for flying R/C models. This is dangerous, you do it on your own risk. This disclaimer of warranty constitutes an essential part of this license. Any liability of the developers of HELI-X will be limited to zero. In addition, in no event shall the developers of HELI-X, or its principals, be liable for any indirect, incidental, consequential, or punitive damages whatsoever relating to the use of HELI-X.

http://www.heli-x.net

Note: Flying with R/C helicopters and planes may be dangerous. R/C models are not a toy.

Have fun with HELI-X.

Contents

http://www.heli-x.net

1 Getting Started

The usage of HELI-X is not that complicated so that it is possible to use HELI-X without reading the user's guide completely. But you should consider the following points:

- At the beginning, you should configure the controller, see Section 2.2.1.

- You also have to assign the functions Idle Up 1 to 3 to a switch of your controller or a key of the keyboard. Only then, the engine of the helicopter or plane can be started.

- Then you can select an airport. There are different kinds of airports available. Use a limited resolution when your graphics adapter is not so powerful.

- 3D-Scenes offer the possibility for a variable pilot position and to fly "with" the helicopter.

- Then, you can select a model and your training can start.

- Many functions in HELI-X can be used by assigning functions to a switch on the controller or a key on the keyboard. Thus, you might want to have a look at Section 3.

- If you are using a Spektrum USB Dongle, only 8 channels are available to assign functions in HELI-X

Have fun with HELI-X!

2 The Menu

Most of the functionality of HELI-X can be accessed from the menu bar.

Simulation	Configuration	Record/Play	Multi Player	Training	HUDs	View	Shortcuts	Options

- **Simulation.** Here you can load airports, helicopters, and utilities. In addition you can load and save scenes. Also all editing processes are started here.

- **Configuration.** Here you can perform many configuration tasks: controller, stick display, camera, view, sound, and shadows. In addition, you can modify the dual rate/expo-settings. Furthermore, the brightness of the scenes and the wind speed can be set.

- **Record/Play.** Here you find everything for the record/play functionality.

- **Multi Player.** Here you find all functionality to practice together via the internet.

- **Training.** Available training modes can be found here.

- **HUDs.** HELI-X offers many head up displays (HUD) which can be selected here.

- **View.** HELI-X offers many possibilities to control the view.

- **Shortcuts.** Some of the most important functions in HELI-X can be found here. The complete functionality can be accessed via the key and button functions.

- **Options.** Define the language. You also find everything which belongs to the help system. Furthermore you can update your model and airport library.

2.1 Simulation

For the selection of airports, helicopters, utilities and scenes, there comes up a window of the following kind:

http://www.heli-x.net
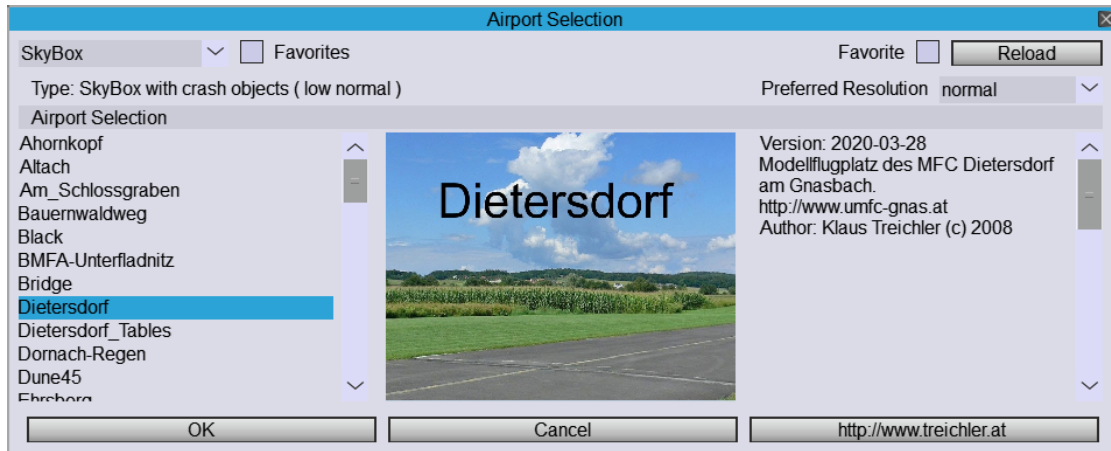
The choice is performed using OK. Note the Favorites and the Favorite check box. You can restrict your choices to the favorites and can define for each item if it is a favorite.

You can also find this combo box:
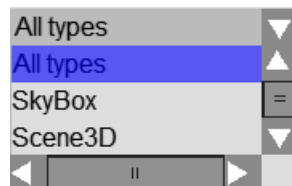
Using this box, you can restrict the choices, see next section.

2.1.1 Airport Selection

There are two kinds of airports in HELI-X: SkyBox,and Scene3D. SkyBox means a photo scene, while in Scene3D you have more possibilities for the camera settings, e.g. variable pilot position or flying "with" the model. Note that you can choose from different resolutions. The better resolution needs more computing power on your graphics adapter.

2.1.2 Model Selection

Selection of the helicopter and plane models.

2.1.3 Utilities

Currently there exist three kinds of utilities: boxes (in particular interesting for landing training), pylons and heliports. The positioning is done using a coordinate system with x and z running at the bottom and y pointing upwards. To simplify the positioning it might help to switch the matrix view on (function "Toggle Matrix View"). It is not possible to remove the utilities piece by piece. To remove all of them together, there is a menu entry in "Shortcuts".

2.1.4 Scene Selection

A scene is an extension of an airport with utilities, other 3d objects. Typically, in scenes there are also tracks defined.

2.1.5 Track

A track (new in V7) describes a sequence of operations which the helicopter has to to. This can be special training tasks or, e.g., an fpv track. When a track is activated, the timer can be used to measure the time.

2.1.6 Start position

You can choose from different start positions. They will also define where some trainings will take place.

2.1.7 Pilot position

You can choose from different pilot positions, but only in airports of type Scene3D.

2.1.8 Colors

If the selected model offers the possibility to change its colors, you can do it here

2.1.9 Recent Airports

For the quick selection of recently used airports.

2.1.10 Recent Models

For the quick selection of recently used models.

2.1.11 Airports Favorites

For the quick selection of the favorite airports.

2.1.12 Models Favorites

For the quick selection of favorite models.

2.1.13 Terminal

Opens a terminal window to show debug information. Usually you do not have to consider this terminal window.

2.1.14 Factory Settings

Bring HELI-X to the factory settings. All user data will be destroyed.

2.1.15 Quit HELI–X

When you quit HELI-X then your last settings are stored automatically. They are used for the next start of HELI-X.

2.2 Configuration

2.2.1 Controller

This is perhaps the most important configuration issue.

Note that in HELI-X there is a configuration for helicopters and one for planes. Thant means you have to select a helicopter, do the configuration, then select a plane, and then do the configuration again.

Starting point is this window:
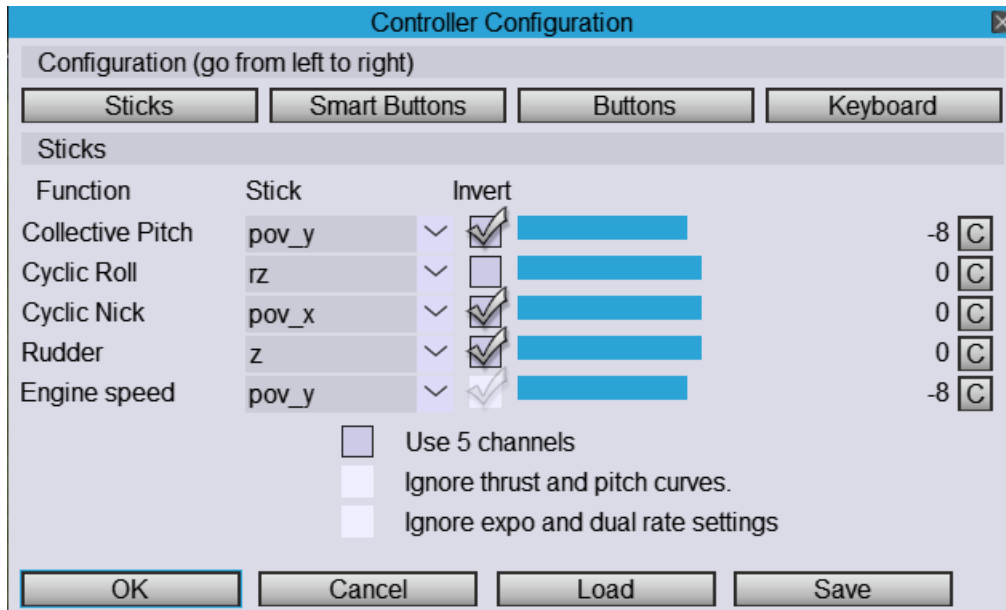
Here you configure (from left to right):

- The calibration and the definition of the stick functions

- With the Smart Buttons feature, you can configure the buttons using a guided method.

- In the buttons configuration you can configure the buttons with a lot of freedom.

- The keys of the keyboard.

The calibration and configuration of the sticks is made here:
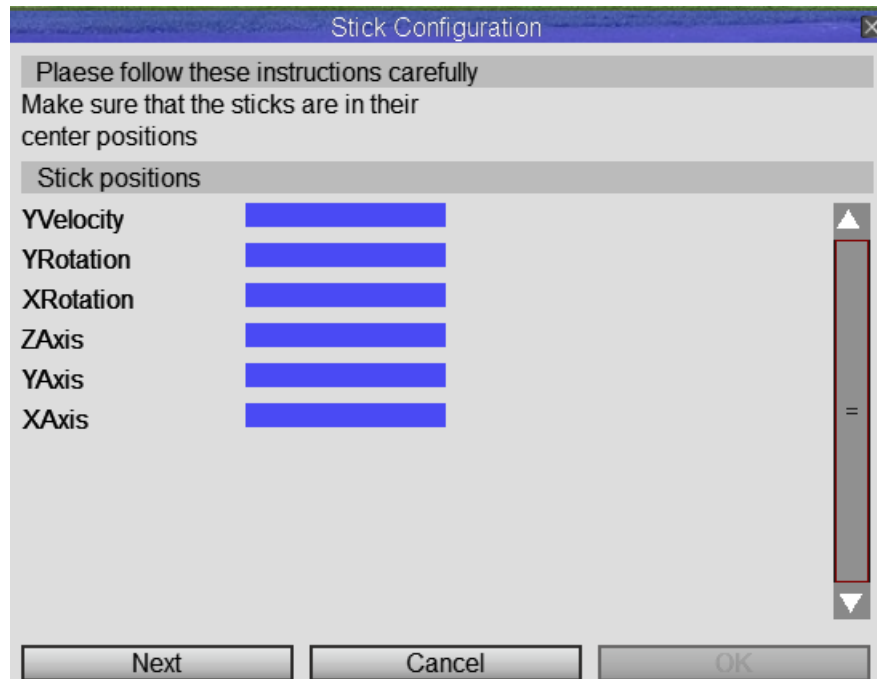
You only have to follow the messages in the window. **If you have other switches or sliders on your controller, do not forget to move them to their end positions during the calibration.**

Note that in the first place, the bars do not move.

The smart button configuration comes up with this window:
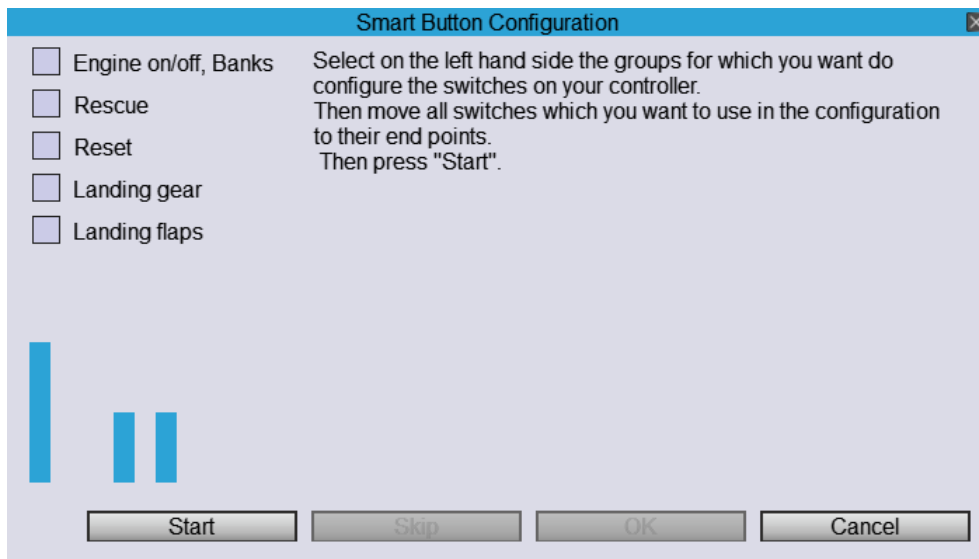
You select with the boxes which function you want to configure, then press start and follow the instructions. You can do this step by step. So in the first step the engine functions; later, the rest of the stuff.

A more sophisticated configuration of the switches can be done by pressing buttons. You can see them in a window of the following form:
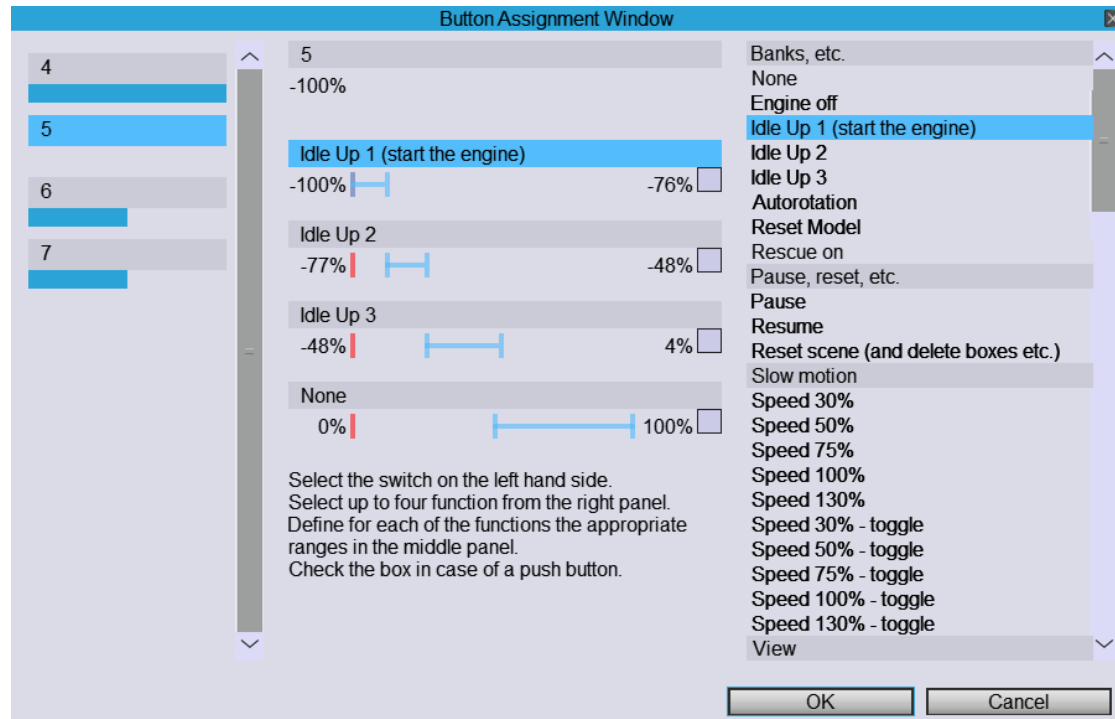
In the left columns you can see a representation of the hardware switches which are recognized and ready for configuration. Select one.

In the middle you can define up to four different functions to the switch. You can move the bars to specify in which range the particular function is active. Then click on the bars in the middle column to activate it and select a function from the right column.

When you have a momentary switch and want it to act as a flip–flop, mark the corresponding check box.

The remaining functions can be assigned to keys of the keyboard. For this, use the following window:
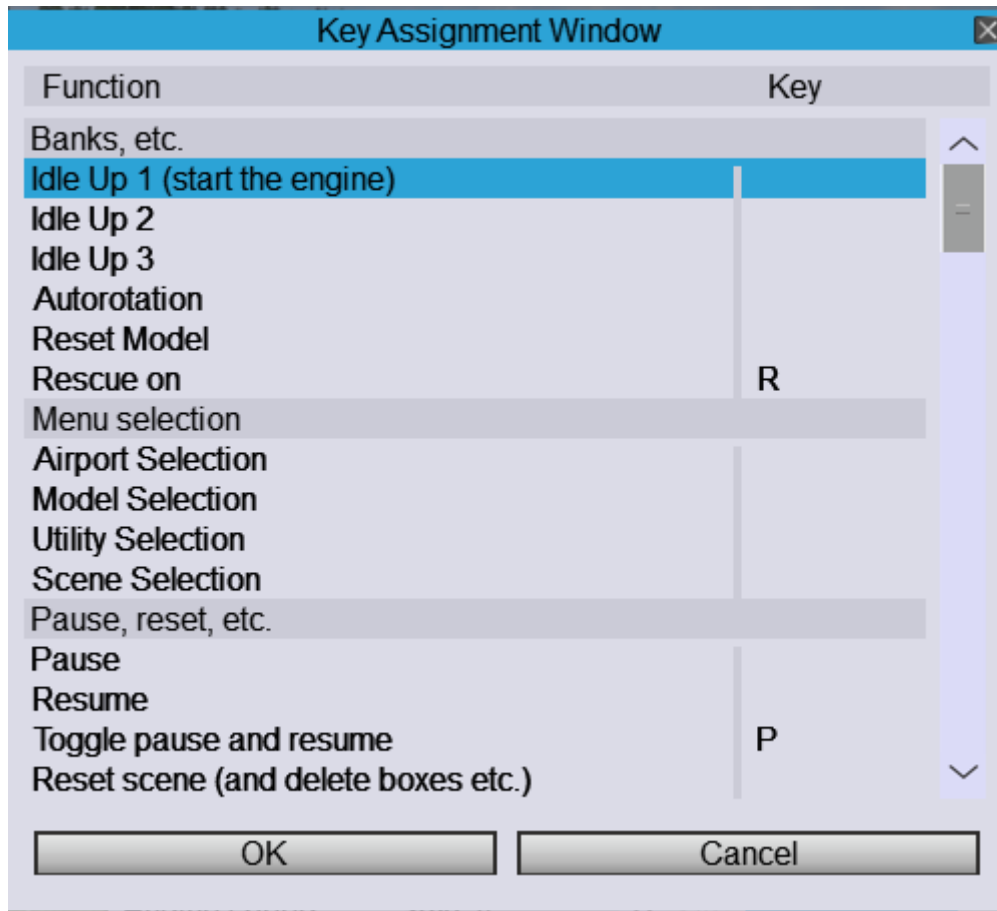
The exact meaning of each function is explained later. It is of crucial importance for the use of HELI-X, that the functions "idle up 1", "idle up 2", and "idle up 2" are assigned to a button or a key. Otherwise, it is not possible to start the engine.

You do *not* have to store the settings. This is done automatically. This also works if different controllers are used with HELI-X. The save function is only to be used, if you want to use different settings for the same controller.

2.2.2 Rescan USB Controllers

You can connect several USB controllers to be used in HELI-X. If you want to change the active controller, select this. Note that, when a VBar Touch is connected, it has priority

http://www.heli-x.net

as long as it is connected.

2.2.3 Stick Display

There is the possibility, that the stick movement of the pilot is shown in HELI-X:
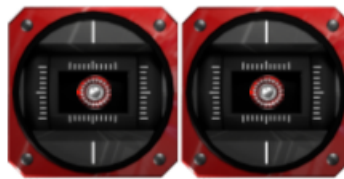
The configuration of the stick display can be quickly done by selecting Mode 1 to 4, or by the configuration process. The configuration of the stick display is done in this window:
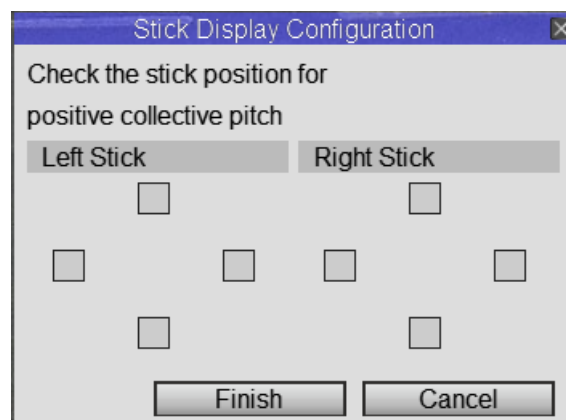
It is independent of the configuration of the controller, so that it is possible, for example, that you fly with mode 1 but the stick display looks as if you flew with mode 2.

Using next color and next stick color, you can change the appearance of the stick display.

Turning on the ghost sticks, you can see your stick movements together with the stick movements of a video model or a remote model in multiplayer.

http://www.heli-x.net

2.2.4 Camera

In HELI-X there are two cameras available. They, and the camera in the VR situation can be configured in this window.
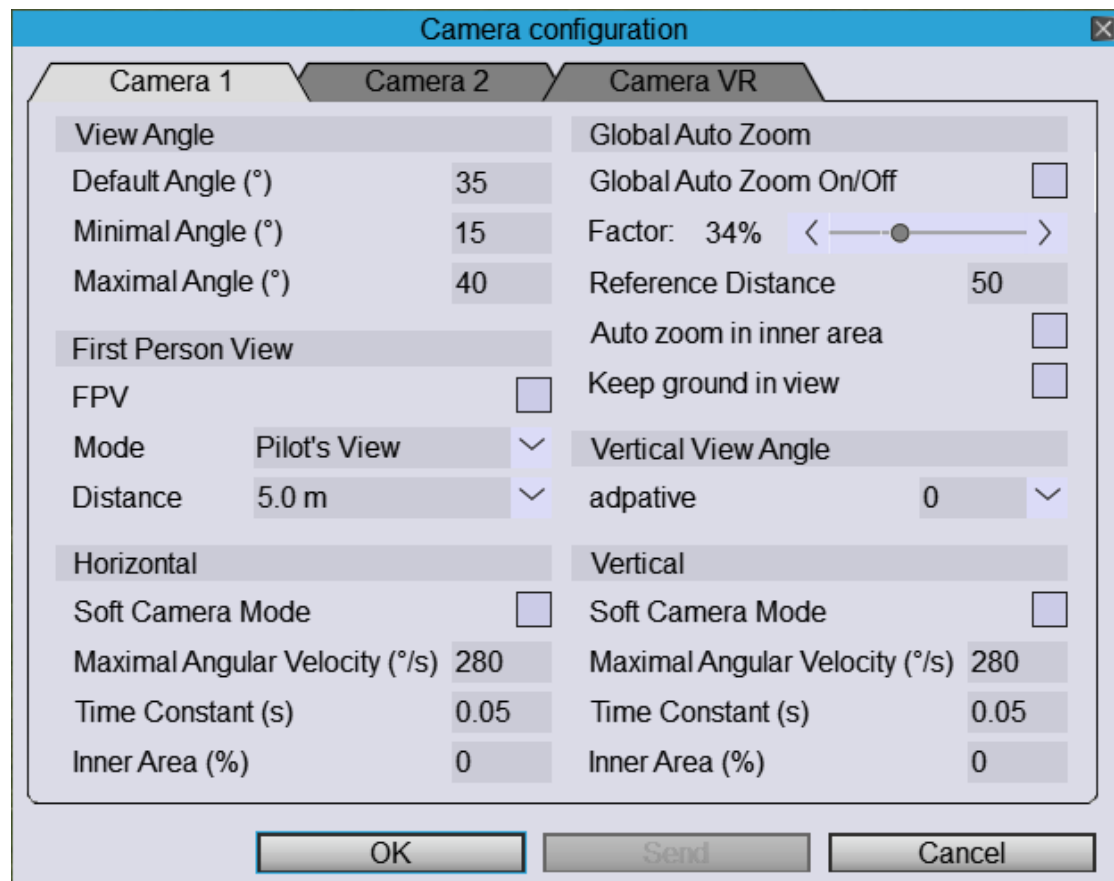
With *View Angle* you define the default as well as the minimal and maximal view angle.

Gobal auto zoom is switched on and off at the top right. Auto zoom means that there is an automatic zoom-in when the helicopter is far away from the pilot and vice versa. The corresponding parameter can be input with the slider. 1.0 (i.e., the rightmost position) means that the zoom is performed in such a way, that the helicopter keeps its size. 0.0 means that there is no automatic zoom. A good starting point for you might be a value 0.2. The reference distance defines the starting distance at which the auto zoom works. If you select "auto zoom in inner area" it works also within a distance which is less

than the reference distance. If you select „keep bottom in view", then it is automatically zoomed in such a way, that the bottom is visible all the time. If you use this options, it might be useful to activate the vertical soft camera.

> Hint: In reality there is no auto zoom. So you might want to try to switch off this feature after some time and try to fly at distances that you can still see and control your helicopter.

Below, you can adjust the soft camera mode for vertical and horizontal camera movement separately. Soft camera means that the helicopter is not fixed in the midpoint of the window but can move freely around the center and the camera does not follow immediately. There are the following settings:

- The maximum angular speed of the camera.

- The time constant determines how fast the camera reacts to the movements of the helicopter. The larger this number is, the slower reacts the camera.

- The inner area determines the area in which the helicopter can be without a movement of the camera.

A variable camera location can be activated with First Persons's View (FPV), which works only in airports of type Scene3D. There are several settings for the camera movement:

- *Pilot's view* means that the camera moves with the helicopter as if you were a pilot inside the helicopter..

- *Leveled Pilot's view* like *Pilot's view*, but the camera is hold horizontally.

- *rigid* means that the camera is moving with a fixed distance behind the helicopter.

- *flexible* means that the camera movement is around the helicopter with a preferred distance.

- *On ground* is similar to the flexible mode, but in this mode, the height of the camera is the usual height of the eyes above the ground.

> Note that the same settings are also valid when you look to another helicopter in multiplayer mode or to the video helicopter. And remember that you can switch the view in multiplayer mode by clicking on the pilot's name in the

pilot window or by using a function.

If you turn on the vertical view angle on, the position of the model on the screen corresponds to the view angle.There are different settings.

Further settings for the camera can be found in the menu "View".

In VR mode, you usually use "Freeze View" from the View menu and the settings in the camera configuration are not used.

2.2.5 Flight mode
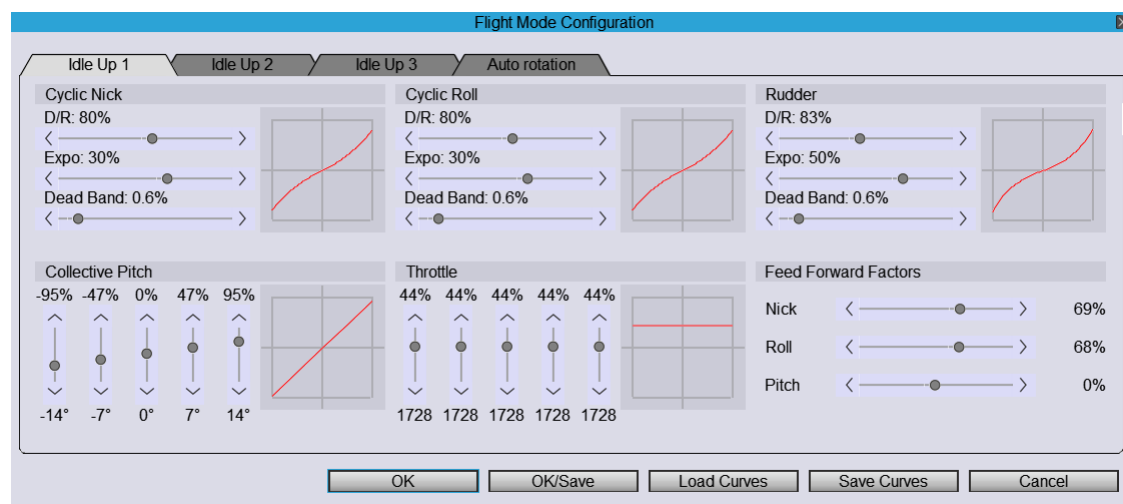
In this window you can adjust the dual rate and expo setting as well as the pitch and throttle curves to your own needs. In addition, you can use the feed forward factors to define the aggressivity, with which the helicopters react to the inputs.

Of importance are the two possibilities to save the settings: If you press OK/Save, then the settings are stored in the parameter file of the selected helicopter. If you press only OK, then the settings are used but not stored together with your helicopter. Thus, it might be a good idea, to experiment with the settings (pressing OK) and at the end you store the settings with the model (OK/Save).

http://www.heli-x.net

You have the possibility to load and save expo and dual/rate curves. The gyro settings are not stored.

You can use the dead band when your controller has problems to deliver a correct zero signal.

For planes, this window looks similar, however some functions, e.g. the feed forward factors are not used.

When you use a drone, you can assign to the diffreent flight modes one of the flight modes "Position", "Altitude", "Sport" or "Acro".

The meaning of these modes is as follows:

- Position: The height and the position is controlled using a GPS sensor.

- Altitude: The altitude is fixed. The stick controls the bank angle of the drone.

- Sport: The altitude is not fixed, the stick controls the bank angle of the drone.

- Acro: Like the sport mode, but the bank angle is not restricted.

2.2.6 Trim
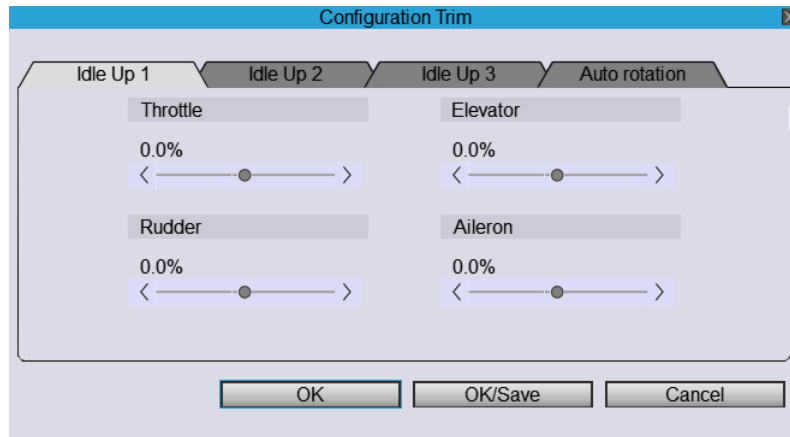
Here you can assign trim values, use them (OK) or store them to the current model (OK/Save). The trim parameters are typically not used for helicopters, but might be useful for planes, in particular when your controller does not have trim switches.

2.2.7 Center of Gravity
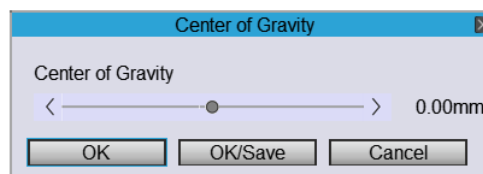
Here you can move the center of gravity for planes.

2.2.8 Wind/Brightness
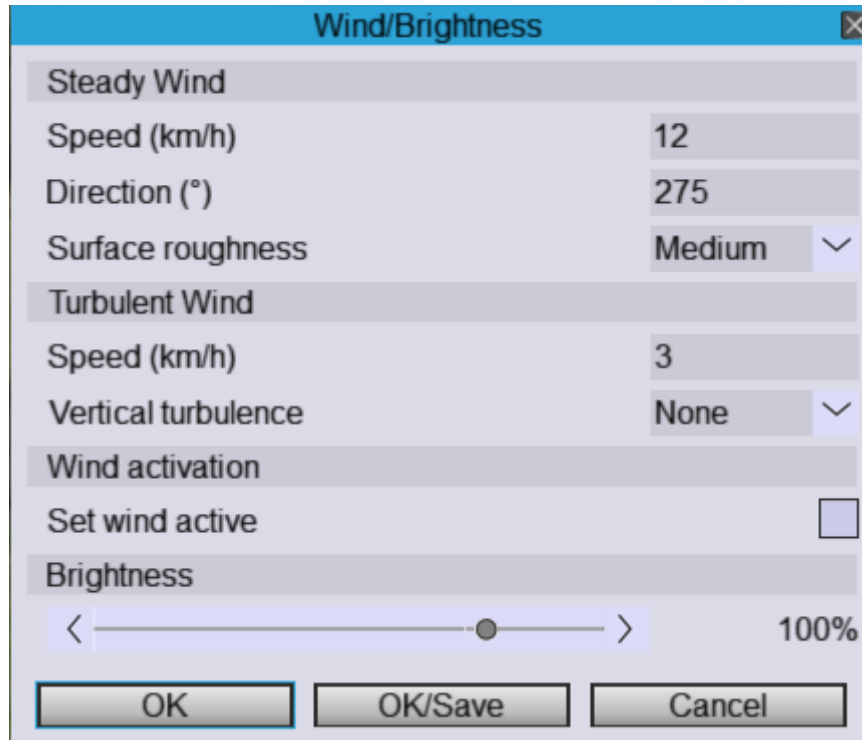
The wind in HELI-X has two components (as in nature):

- The steady or laminar part has a constant speed and direction. The surface roughness determines the shape of the bounary layer of the wind speed.

- The turbulent part changes its direction and speed during time. The strength of the turbulent wind is determined by a typical speed. In addition you can specify whether the wind contains vertical turbulent components. With the checkbox you can switch the wind on and off. There is also a function which can be assigned to a key, if desired.

In some airports, in particular in those which can by used for hang soaring, the wind settings are stored in the airport settings.

The brightness of the scene can be influenced with the slider at the bottom. This is interesting for night flying. In this case you should use helicopters which have lights.

The standard value is 90%.

2.2.9 Slow Motion

Here you can choose the simulation speed (with the same physics) of your own helicopter.

2.2.10 Rescue

There is a rescue mechanism included in HELI-X. Here you configure it: You decide

- the speed vs. robustnes of the control algorithm.

How the rescue function is activated can be defined in the controller settings.

2.2.11 Global Settings

In this window you can adjust HELI-X to the possibilities of your hardware.
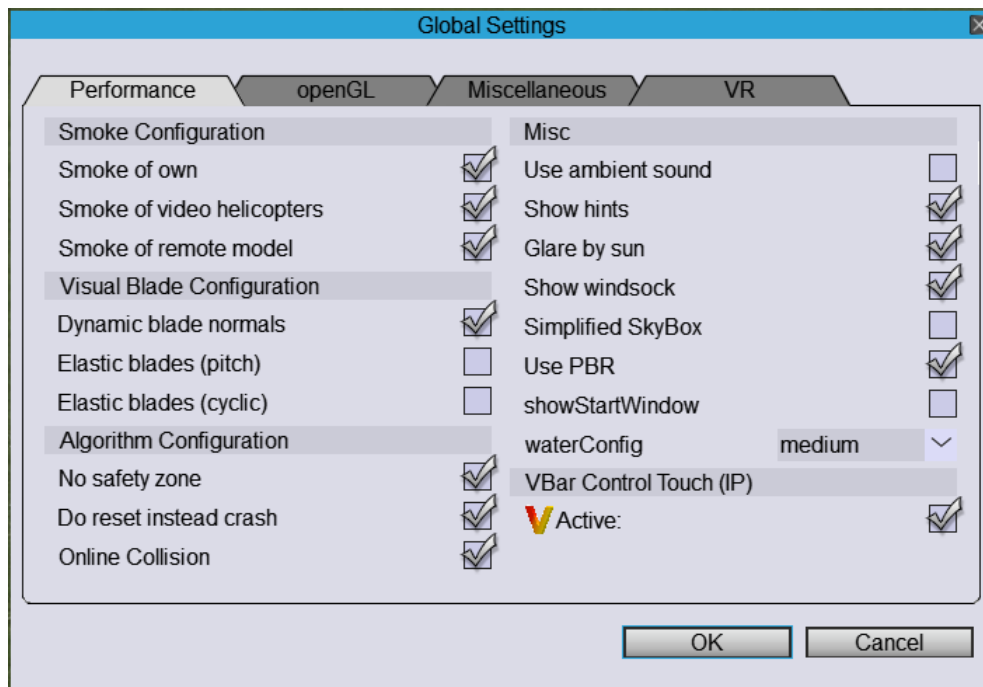
In the performance tab, you can configure the smoke settings. In the visual blade configuration you decide, if dynamic blade normals are used so that the reflections of the rotating rotor depend on the actual pitch value.

In some helicopters, the blades and/or the paddles are elastic or movable. With these switches you decide if the elastic parts react to collective and/or cyclic inputs. The default value is of course on. Turn the switches off if you run into performance problems.

Furthermore, you can deactivate the safety zone and you can disable the crash simulation so that the simulation starts faster after a crash. Decide also, if your helicopter will be crashed when it collides in a multi player simulation with an other helicopter.

Some airports provide an ambient sound file. Decide if you want to use them.

On some older machines, the hints are bad for the performance. Choose if you want to use them.

You decide if you turn on the glare by sun and if you want to show the wind sock if the airports defines it.

Simplified sky box means that the upper and lower picture of the sky cube are used in a lower resolution. By this you can reduce the computer power used to render the scene.

Use PBR stand for the usage of physical based rendering. This is a new feature in V10 which allows a better rendering of models.

In particular on MacOS we use a start window where you can define a mouse factor which is important since MacOS does not allow HELI-X to find out what is the current screen resolution. Once you have found the correct factor, you can avoid further pop ups of the start window.

Also new in V10 is the rendering of water in some scences. here you can define the internal resolution, again a compromise between performance and render quality.

If you want to use a VBar Touch using a WLAN connection you have tu turn it on here.

The second tab provides possibilities to adjust the OpenGL settings.
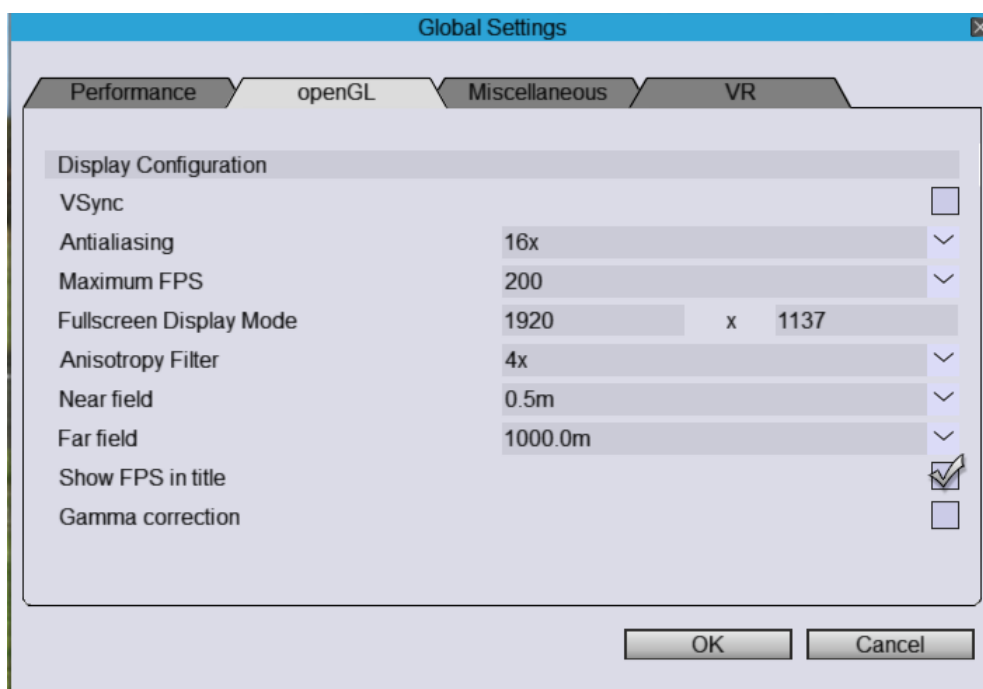
The anti aliasing settings are responsible for the smoothing of edges. Theses settings might drastically influence the performance of HELI-X, depending on your graphics

adapter. VSync means that the frame rate is synchronized with the monitor refresh frequency, which causes a limitation of the frame rate. If you do not limit the frame rate, HELI-X tries to get as much performance as possible to maximize the frame rate of the simulation. The resolution settings show the display modes of your graphics adapter for full screen mode. You might have to try different settings to see what works best on your hardware.

Near and far field define the borders in meters where the range is where the scene is rendered.

In addition you can decide if the current FPS is shown in the window title and if you want to use a gamma correction.

In the third tab you can find further settings:
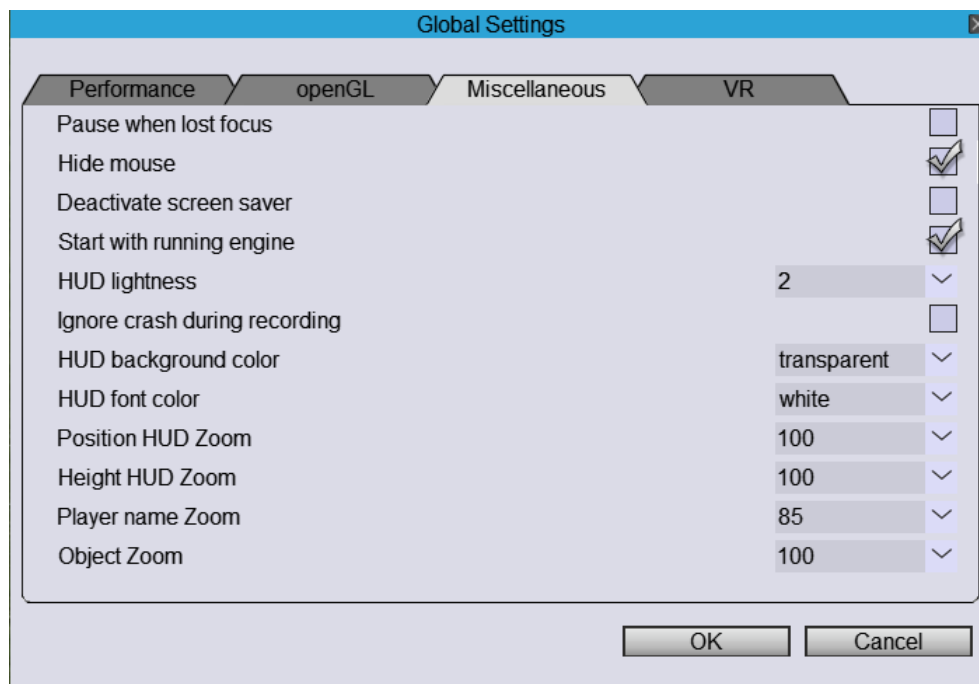
- You decide if HELI-X should go into pause mode, when its window looses the focus.

- You decide whether the mouse pointer should dissappear automatically after a few seconds.

http://www.heli-x.net

- When you want to deactivate a screen saver, you can mimic a small mouse movement every few seconds.

- You can start with a running engine so that you do not need to turn on flight mode 1 or 2 after each crash.

- And you can define the brightness of the Head-up-display (HUD).

- You can ignore crashes during recording so that the recodring does not stop.

- And you can define the appearance of the text HUDs (background and font color).

- You can define the zoom level for the position HUD and the height HUD.

- You can switch on in multiplayer mode that the names of the players are displayed. Here you can define the size of the font.

- Last but not least, you can scale your model to make it generally larger or smaller. Note that this is not realistic.

In the last tab you can find settings for the VR mode:
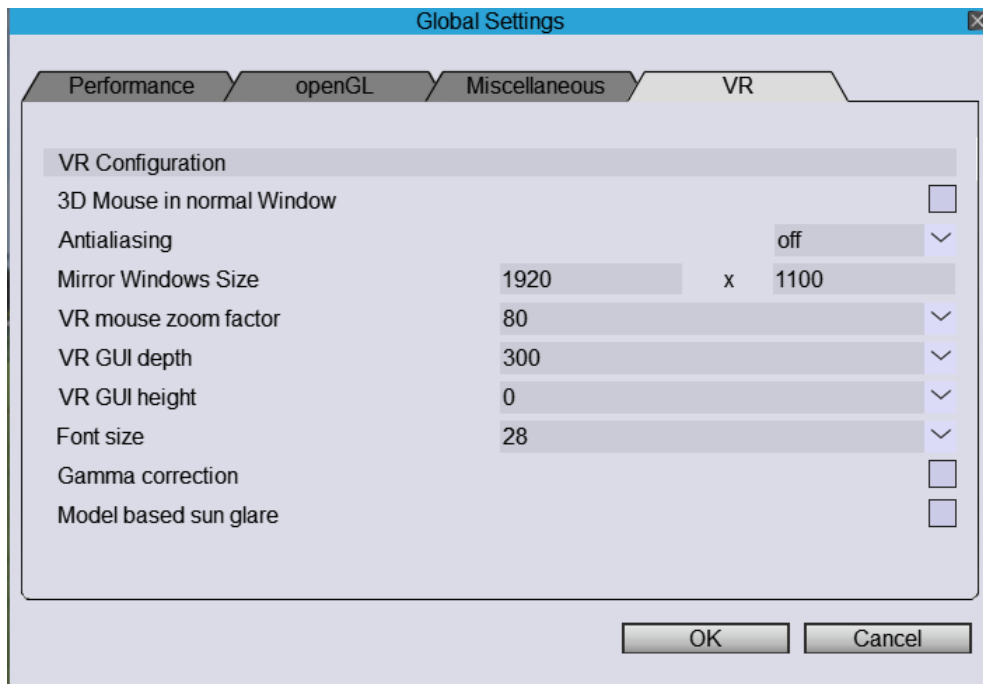
The gui in the VR mode works as follows:

- In VR mode every settings is made in the VR scene.

- The so called mirror window is used to transform the mouse movement of the screen to a mouse movement in the VR szene.

- Thus, you have to take care that the mouse pointer on the computer screen stays in the mirror window.

In particular, you can define the following items here:

- You decide if the 3d mouse is also shown in the non VR mode. You can use this to check if the mouse factor is correct.

- The anti aliasing factor is the one for the VR. It might be possible to define Anti aliasing also in the configuration of the graphics adapter.

- The mirror window size should be as large as possible in your screen.

http://www.heli-x.net

- the mouse zoom factor defines the size of the mouse in VR

- The gui depth defines how far away from your eyes is the gui in the VR mode, and thus the size is defined.

- The gui height can shift the gui in the VR scene in vertical direction.

- The font size for the gui in VR can be defined here.

- You can turn on Gamma correction.

- In VR it is not obvious if the sun glare should be based whether if your view is directed to the sun or if the model's position is in direction to the sun. You can change this here.

2.2.12 Sound

Here you can turn on and off the sound and adjust the volume. The Doppler factor influences how the pitch of the sound changes dependent on the velocity of the helicopter.

You can reduce the sound of remote models in multiplayer sessions. You can turn on transmitter sounds

You can also define the volume of the ambient sound which is available for some airports.

2.2.13 Shadow configuration

The calculation of the shadows (in particular when there are many utilities in the scene), can influence the performance if you have a graphics adapter which is not high performance. Thus you can define here which shadows are displayed.

2.2.14 Rotor transparency

Define the visibility of the rotation rotor disk.

http://www.heli-x.net

2.2.15 Canvas size

You can set the canvas size to a standard value.

2.3 Flight Recorder

The flight recorder has the following features:

- A recording can be started and stopped with the menu entry "record/stop". The recording time is then displayed in the status bar. In case of a crash, the recording is stopped automatically. After recording, the video is in the internal memory of HELI-X and can be played.

- To play the video you can use the menu entry "Play/Stop" or you can use the play control window:
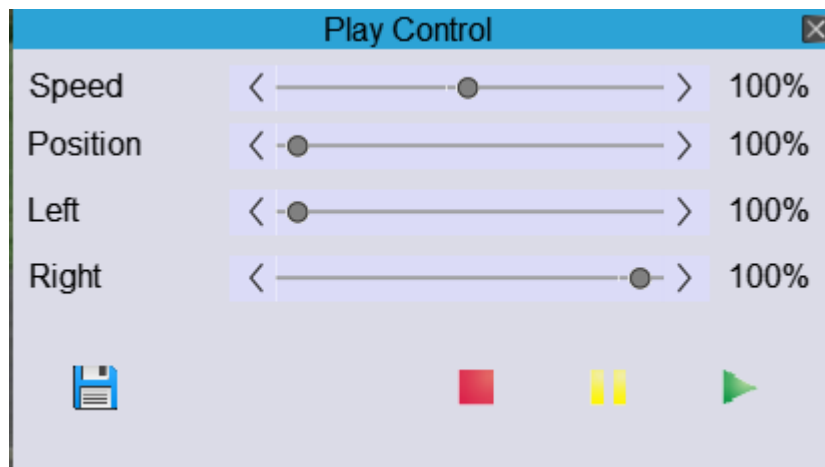
It is also possible to change the position of the video and you can define left and right markers where the movie is repeated. It is also possible to save the selected video snippet.

- It is possible to fly simultaneously with the video. Using the menu "View Direction", it can be determined to which helicopter the camera looks.

- There is only one video stored in the internal memory of HELI-X. If you record another one, the previous one is lost.

- To store a video, use the menu "Save Flight". This creates automatically a new directory to store the video.

- If you want to share this video with you friends, make a zip file out of the complete directory. You find it in `resources/flights`.

- You can edit the corresponding xml file to include your comments about the video.

- You can also add a picture `preview.jpg` to the directory. Then this picture is shown in the video selection window.

- You can access stored videos with "Load flight".

- With „Export Movie" a video can be stored in the format mp4, avi or as a sequence of pictures. Here you have the best quality, but the export needs some time. The sound will not be recorded in the file.

2.4 Multi Player

The multi player functionality is based on a client/server principle. Typically several servers are already running. In the demo version there is also included a program so that you can run your own server without graphics. If you want to create your own server, one pilot can act as a server: He starts the sessions and predefines the airport and the utilities. The other pilots can connect as client. All communications are directed via the server. Thus, the server should have a good computer with a fast internet access. The multi player menu looks as follows:
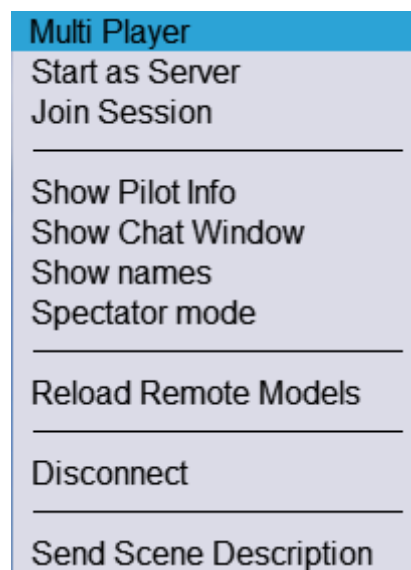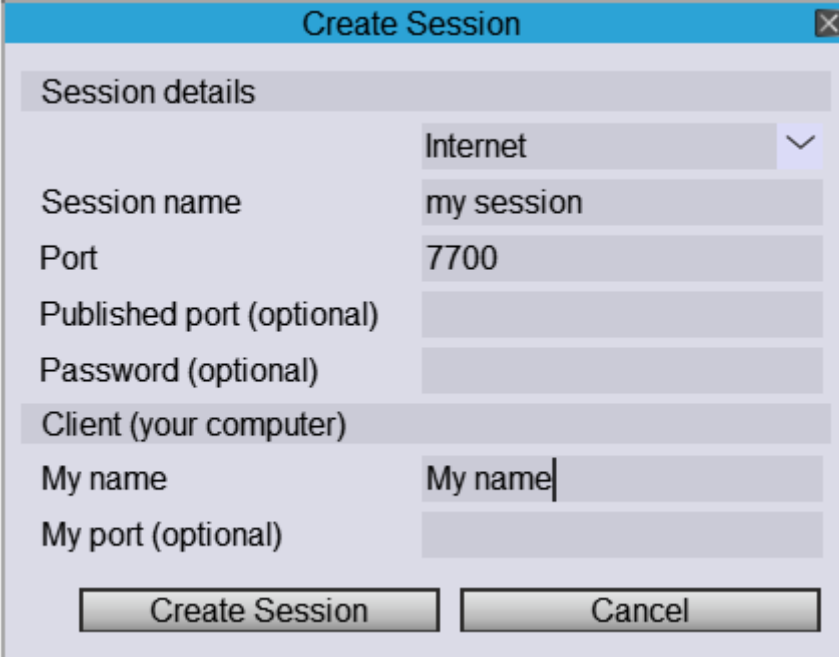
In this state there are two possibilities: You can start as a server or you can connect as a client to a running session.

2.4.1 Create session as server

To create a session the following window appears:

- Before creating a session you have to select the airport and the utilities.

- In the combo box in the top of the window, you decide if this is a session used in the internal network or a session via the internet. Depending on this selection, the IP address for internal or external use is published.

- Then you specify a name for this session.

- The port defines the port to which the clients have to connect. It is published.

- If you use a port mapping, you can define the port number which is published to the others.

- If you want, you can define a password. In this case, only those people can connect who know the password.

- You have to specify your name which is also visible to the other players.

- You can specify a port on which you want to send. In most cases the field can be left empty.

- Pressing the right button, the session is created.

- And then, other pilots can connect to your session.

2.4.2 Connecting with a running session

To connect with an existing session you can use this window:

Name	Airport	Pilots	IP	Port	Mode	Private
Heli-X Users	Dietersdorf	0	83.135.172.168	65530	TCP/UDP	no
Helischule Gonzalez	StPeterOrding-Stra	0	83.135.172.168	62500	TCP/UDP	no
Belarus-Team	Dietersdorf	0	83.135.172.168	65200	TCP/UDP	no
Heli-X Demo	Dietersdorf	0	109.192.122.238	62500	TCP/UDP	no
Heli-X Training	Schorndorf	0	87.180.219.104	62499	TCP/UDP	no
Flybar UK	Dietersdorf	0	87.180.219.104	62500	TCP/UDP	no
Helischule-Gonzalez	Nordsee	0	84.118.114.23	62500	TCP/UDP	yes
IRCHA Heli-X Server	Dietersdorf	0	173.231.213.37	62499	TCP/UDP	no

Server		Client (your computer)	
IP	83.135.172.168	My name	Michael
Port	65530	My port (optional)	
		Password (optional)	

Connect   Cancel

- In the top area you find a list of all sessions HELI-X is aware of.

- If you click on a session, then all the necessary parameters are put into the right places.

- You need the port and the IP address of the server as well as the kind of the connection.

- If you want to connect to a session which is not in the list, you can write the data in the fields by yourself. In this case, the pilot of the server has to give you the information.

- You have to input your name and you can input a port which is the send port of your computer. In most cases you do not have to determine a port.

- If the session has a password protection (you see this at the column "private" in the list) you have to input the correct password to join the session.

- If you press "connect", the connection to the server will be established.

2.4.3 Other windows

The window with the pilot information lists all pilots together with their helicopters. When a pilot selects a model which is not available at your computer, you get an appropriate message in this window.

- If you click on a line, the view follows automatically the particular pilot.

- With the check boxes you can supress other helicopters or chat messages from a pilot.

- If the box "Show on changes" is checked, the window is brought to the front at every change. This might be an advantage when you want to know who is in the session, but it might also disturb you during your flying.

- The server has an additional button to disconnect another player. You have to select the appropriate player in the list beforehand.

Another window which can be activated is the chat window. It's use is obvious. A message is sent when the send button is pressed or when you type Enter. The messages which were sent before you connect, are not shown.

2.4.4 Other functions

You can define if the names of the pilots of the remote helicopters are displayed. Note that the font size can be chosen in the global settings.

In the spectator mode, the view is automatically moved from one player to another.

If the session is out of sync, or you have updated your helicopter library and you want to use the new helicopters for the remote pilots, you can use the menu entry "Reload Remote Helicopters". In this case, all remote helicopters are removed and reloaded

afterwards. You should need this item only rarely. The last menu item can be used to disconnect. If you act as the server, the whole session is closed.

2.4.5 Requirements for the communication

The communication of the HELI-X multi player mode is based on TCP and UDP. Please note the following hints:

- The client has to allow outgoing connections. For this, a configuration in the router might be necessary. Used is the given port and the two following port numbers, for example 1025-1027.

- The port should be able to communicate with TCP *and* UDP.

- The server has to allow incoming connections.

- If you have a fire wall, you should be aware of the parameters there. In Windows, it might help to deactivate the fire wall.

- If your router does perform a port mapping, it has to be configured accordingly.

- All these settings depend strongly on your particular hardware. Thus, if there are any questions, you should ask for help at the HELI-X forum http://forum.heli-x.info.

2.5 Training

Here you can start and configure all the available training methods.

http://www.heli-x.net

We divide the different trainings in trainings for helicopters and trainings for planes:
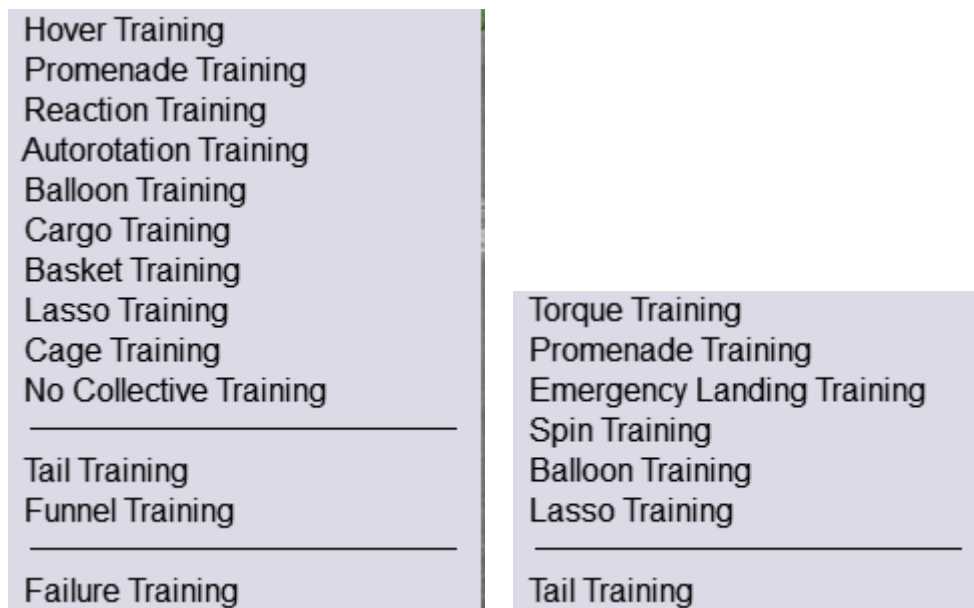
The first entry in the menu allows to start and stop the last used training.

Hover Training
To configure the hover training, the following window appears:
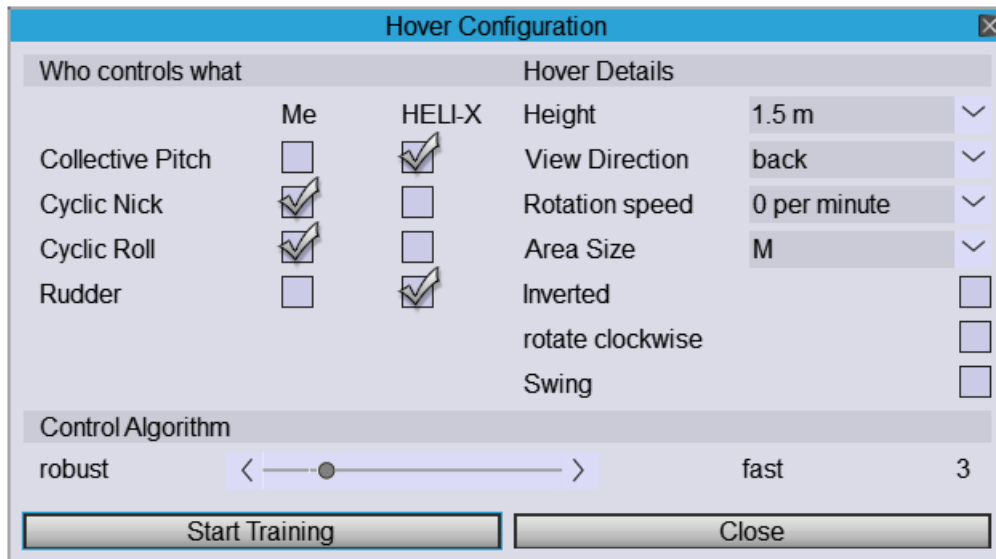
- On the left hand side, you can decide which functions are controlled by HELI-X and which you want to control by yourself.

- Several details are specified on the right hand side.

  - Do you want to practice inverted or not?

  - At which height is the hover training performed?

  - Which side of the helicopter should be visible?

  - With the box "Rotation Speed" you can define if the helicopter should rotate around the vertical axis or not. You can also specify the rotation speed.

  - With the checkbox below you can change the rotation direction.

  - And Swing means that the heli makes a rotation around the axis from -90 to 90 degrees.

- Those functions which are controlled by HELI-X, are performed in such a way, as

a teacher would do. The position of the helicopter is therefore not held artificially. The control algorithm can be set to fast or robust according to the value of the slider. If the helicopter starts to swing you should choose a lower value. And if the heli does not hold its position, chose a higher value.

- Using the buttons, you can start or stop the training.

- It is possible during the training to change the airport or the helicopter.

Promenade Training

The promenade training is similar to the hover training. However, this time the pilot walks around the helicopter. The nose of the helicopter should typically point into the same direction. The promenade training does only work when an airport of type Scene3D is selected.

Reaction Training

The idea of this training is that the helicopter starts in a random situation and it is your task to bring the helicopter safely to the ground.

Autorotation Training
For the configuration of the auto rotation training, the following window appears:
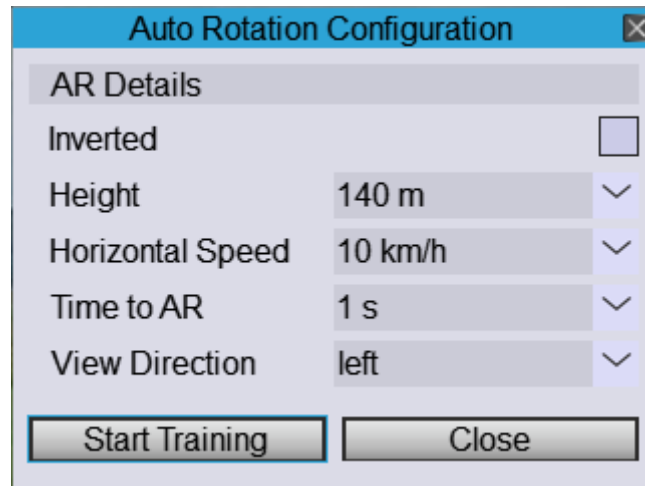
Here you can define the height for the start, the horizontal velocity during start, the time until the engine stops and the view direction to the heli. The checkbox defines if you want to start inverted. The goal is of course, to land with a small vertical velocity.

The stall meter might help you with the training. It is explained later.

Balloon Training
In order to configure the balloon training, the following window is used:
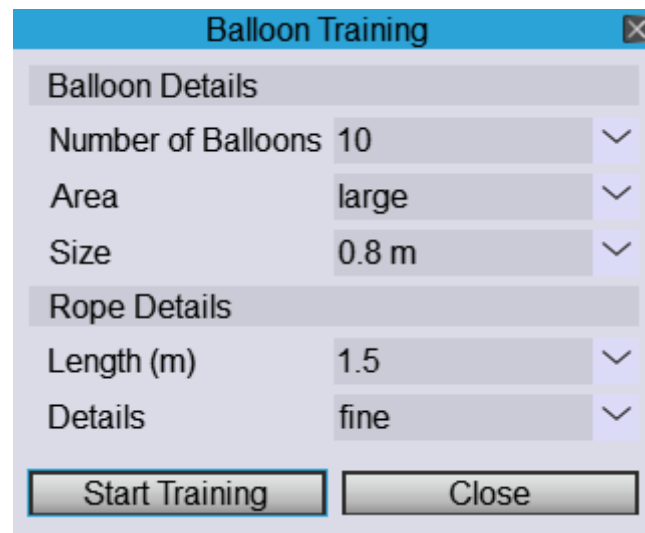
You choose the number of balloons, the size of the area where they are and their size. For the lines, you define their lengths and the details in the dynamic simulation.

Cargo Training
For the configuration of the cargo training, you can use the settings in this window:

The settings are similar to the balloon training. Note that you have to move the cargo from where it is placed to the active heliports. There it is unloaded automatically as soon as it touches the active heliport.

Basket Training
The basket training is similar to the cargo training. But the balls are stored in baskets.

http://www.heli-x.net

Lasso Training

The goal of the lasso training is to fly with the rope. The rope can be cut by the rotors. So typically the helicopter has to fly backwards.

Cage Training

The goal of the cage training is to hover as long as possible inside of the cage. Depending on the level, the cage moves. If the display of the timer uses too many resources on your computer, you can switch it off in the configuration menu.
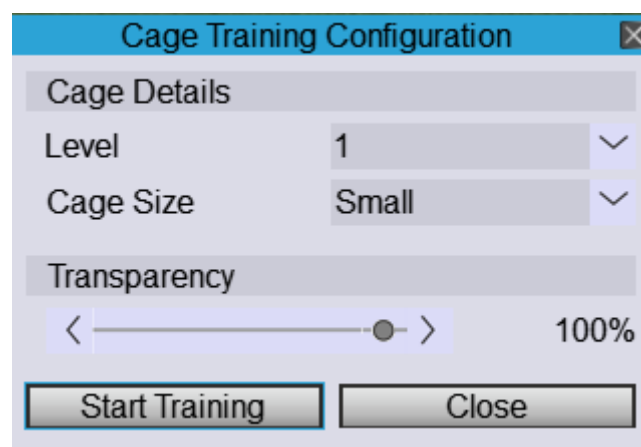
No Collective Training

The idea of this training is that the helicopter is held artificially at a given height and the collective input of the helicopter is ignored. This can be used to practice the stick movement of the cyclic stick, e.g. to get clear about what to do in a piroflip.

Tail Training

The tail training is to practice the tail in such a way, that the helicopter is always oriented along its flight path. For this, the slip meter shows the angle difference between the helicopter orientation and its flight direction.

http://www.heli-x.net
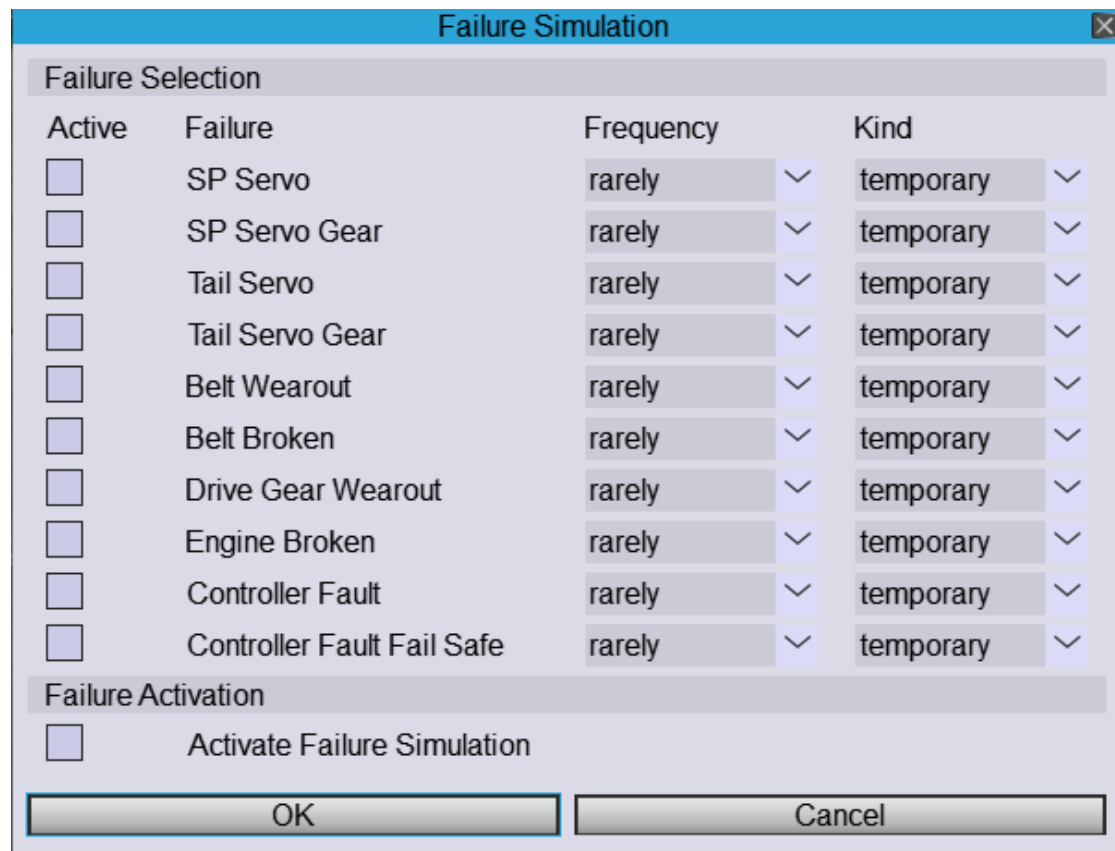
When you start the tail training you get an explanation and the slip meter is activated. Therefore there is nothing to stop.

Funnel Training

The funnel training is similar to the tail training. This time, however, the flight direction is perfectly orthogonal to the helicopter orientation. The display is then called a side slip meter.

Failure Training

With the failure training you can practice the right reactions when parts of the helicopter fail. Use this window to choose the right settings:

Torque Training   Similar as the hover training but to practice to torque with planes.

Promenade Training   for planes, is a torque training where you walk around the torquing plane. Can only be used in airports of type Scene3D.

Emergency Landing Training   Training to land after an engine failure.

Spin Training   In this training, the falling of a plane is only visible in the instruments, but the plane is visible at a constant height. With this you can practice spins without loosing height.

Balloon, Lasso and Tail Traing   for planes is similar as for helicopters.

Crash and Retry Training

When this training is active, after a crash the helicopter is brought to the position some seconds before the crash. So you have a second chance.

Replay Training

The replay training allows that you can repeat your flying starting from a kyboard press or automatically when you crash. You can also define a keyboard shortcut to store the last sequence as a HELI-X movie.

Music Training

You can select an mp3 file so that you can practice your flight to music.

Show Statistics

The statistics window gives you information about your usage of HELI-X.

StickMover

You can use a StickMover. The configuration of the StickMover uses the configuration of the stick display. That means that you can use any flight mode for the StickMover.

2.6 HUDs

The different head-up-displays (HUDs) can be switched on and off in this menu.

HUDs	View
Text HUDs	►
Position HUDs	►
Show HUD	
Show Simple HUD	Show RPM
Show sticks	Show pitch
Show Altimeter	Show Bank Angle
Show indicated air speed	Show Slope Angle
Show Ground Speed	Show Altimeter
Show Variometer	Show Ground Speed
Show G-Meter	Show Flight Mode
Show stall meter	Controller Name
Show slip meter	Show Status Line
Show side slip meter	Toggle status line
Show wind display	Show timer
Show stall warning	
Show flaps	Position
Show sticks (4 in 1)	Position (Afterglow)
Show sticks (moving)	Height
Next size moving sticks	
Show Detail Window	
Show Keyboard Definitions	

A global head up display can be showed. There is also a simplified version available called Simple HUD. The stall meter has three needles. You can observe the speed of the main rotor, what percentage of the main rotor is in stall and how much pitch is given at the tail rotor. The stall meter should help you to practice auto rotation landings.

http://www.heli-x.net

The side slip meter indicates of the helicopters orientation is correct regarding the speed. The same display is available when you fly sidewards.

The timer can show different times depending on the state of HELI–X. During a training, the required time for the training is displayed. When a track is active, the track time is displayed. Besides the current time, you can see the last best times below the current timer.

The heads up displays also shows the flaps settings, the retractable gear status as well as a stall warning, which shows the stall status at the profiles for the left and right wing, elevator and rudder. There are also text huds and position huds. Note that the zoom for the position huds can be defined in the global settings.

New are the moving sticks which is a stick display which stays at the helicopter (usefull for VR mode). You can change the size of these moving sticks with the menu entry above the line.

2.7 View

In this menu you find several functions to define the view within HELI-X.
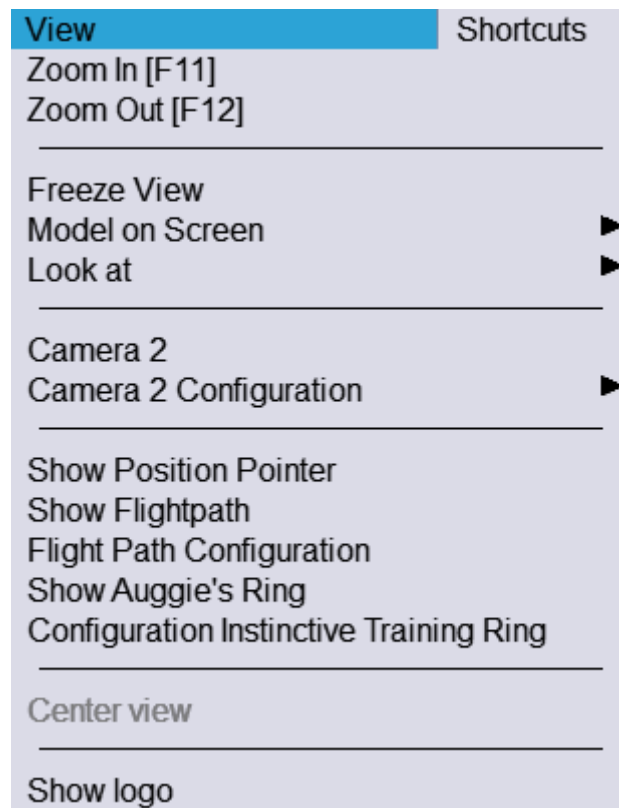
- Zoom functions (instant zoom) are set to F11 and F12 (can be changed). In contrary to the zoom functions they are active while pressing the keys and not while releasing the keys.

- Freeze view means that the view does not follow any helicopter anymore. You can look around using the arrow keys. This is the standard in VR mode. If you deactivate them in VR mode you might get unwanted results.

- Using „model on Screen" you can achieve that the helicopter is above the center of the screen. This may help to see the ground earlier during auto rotation landings or to achieve to see both the cargo and the helicopter together during the cargo training.

- With the menu item "Look at" you can decide which helicopter you want to follow. This can be your helicopter (which is the default) the video helicopter. within an internet session. You can follow a remote helicopter by clicking on a line in the pilot information window.

- With the menu item "Camera 2" you can switch on and off the second camera, which is configured in the second tab in the window camera configuration.

- The menu item "Camera 2 Configuration" is a copy of the first menu items for the second camera.

- The position pointer is a circle on the ground. It may help you with the cargo training.

- The flight path can be illustrated using balls and arrows.

- With the next menu item you can start the flight path configuration.

- Auggie's ring can be activated and configured here. You will find videos from Auggie in the internet which explains how this tool can be used for your training.

- In VR mode, you can user "Center View" to set the default view direction to the current position of the model.

- And the last menu entry permits turning the logo on or off.

2.8 Shortcuts

Some of the most important functions are available in the shortcuts menu. Note that all functions can be assigned to a key on the keyboard or to a button of your controller.
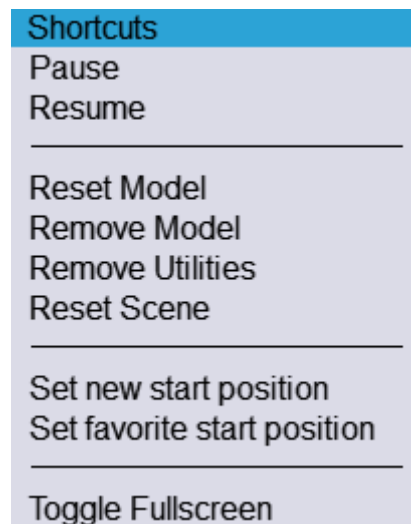
- The Pause and Resume functions stop your own helicopter and release it again. Remote models or the video model will continue their flight.

- Reset model means that your own model is moved to its start position.

- You can remove your own model with the next item. This can be helpful, when you are watching at a video helicopter or in a multi player setting.

- With Remove Utilities you can remove all boxes, pylons and heliports from your scene.

- With Reset Scene you remove your helicopter and all utilities.

- You can set the current position to be a new start position.

- If you set the current position as your favorite start position, this position is stored in the airport configuration and will be accessible in the simulation menu.

- Finally you can toggle between full screen and windowed mode.

http://www.heli-x.net

2.9 Options
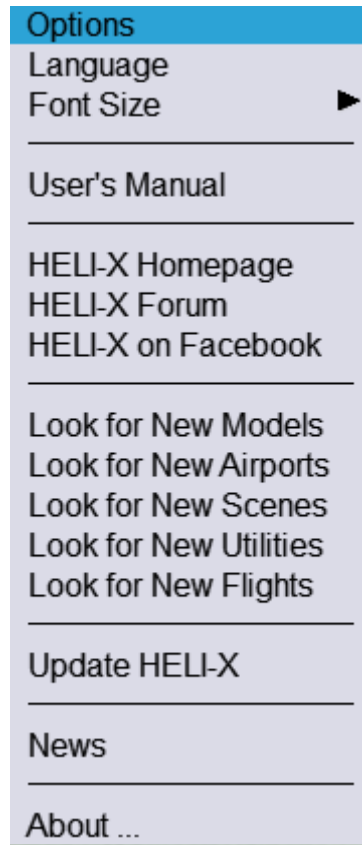
Here is the place to change the language settings and the font size of HELI-X. These settings are stored so that they are effective at the next start of HELI-X. In addition, you have access to the users manual and to some internet link. Using the „Look for ... " entries, you can download new helicopters, airports and so on. As an example, we have a closer look at the window for the download of airports.
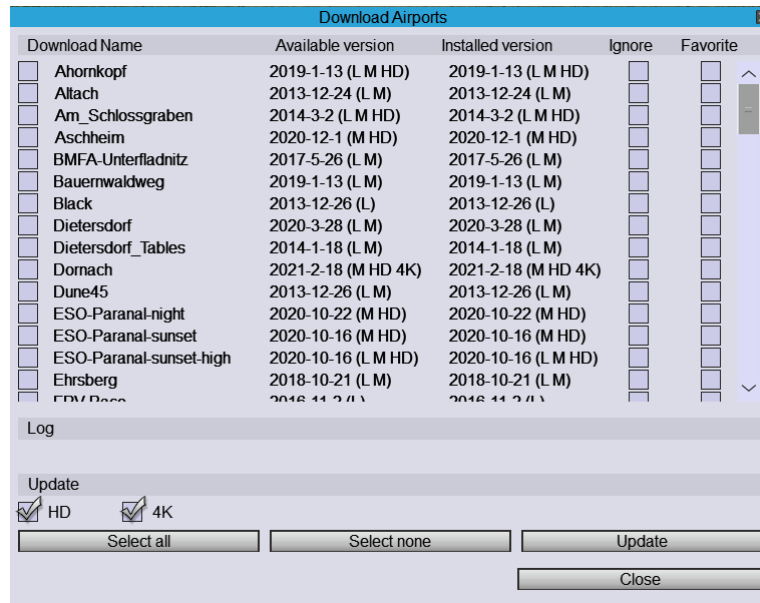
Suggested are files where there are newer version, but only then when the ignore check box is not set. In addition you can mark entries as favorite. This helps in the selection windows. Since airports with the HD resolution (and even more 4K resolution) need a lot of memory and download time, you have to mark them, if you really want to download them. The function „Update" checks if there are updates available.

The news screen gives you the most recent information about HELI-X.

For further assistance have a look at the HELI–X–Forum (http://forum.heli-x.info).

3 Functions

This important section describes the many functions of HELI-X which can be assigned to a button or switch of the controller or to a key of the keyboard. Since many features of HELI-X can be used with these functions, you should have a look at these functions.

Here only a few comments:

- Banks, etc.
  Note that you have to assign the Banks idle Up 1-3 to a switch or a key so that HELI-X will work as expected.

- View
  Note that the function „center view" set the main view in VR mode to the current position of the model.

- Developer
  The function „reload model" loads the current model again. This is in particular helpful when you edit the physical parameters of the model.

4 Virtual Reality

Starting from V10 HELI-X offers the possibility to simulate helicopters and planes in a virtual reality situation using vr glasses. Here are some comments:

- Currently VR works only on a windows system.

- HELI-X uses SteamVR, which therefore has to be installed.

- You start the VR version by another start icon called HELI-X_VR

- Then SteamVR should be activated automatically.

- You control the VR version only through the glasses. Do not try to use the mirror window.

- In the VR glasses you have a mouse which works as the mouse in the normal version. The whole control of the VR system works with a gui which can be used with the VR glasses.

- In order that this works, the mouse movements inside the mirror window are used to control the mouse in 3d.

- Thus, take care that the mirror window has an appropriate size. You set the size of the mirror window in the global settings window. It is a good idea to set the size of the mirror window to fill your screen.

- HELI-X and the VR version use the same configuration files. Thus, it does not matter with which version you perform the configuration.

5 PBR materials

From Version 10 on, in HELI-X, the so-called physical based rendering (PBR) is possible. However, it is not clear that every user wants or can use this because of performance issues. Furthermore, the Wavefront standard does not support PBR. Thus, in HELI-X there are special extensions in the mtl–files. The philosophy is: When the user does not activate „use PBR", then the materials use the old standard.

Workflows in PBR PBR materials offer two workflows, the so-called metallic workflow and the dielectric. Both are available in HELI-X.

5.1 Metallic workflow

The basic metallic workflow uses these parameters in the material description:

- Kb r g b (a)

- map_Kd filename

- metallic x

- roughness y

Kb defines the base color. The alpha value is optional. Ka, Ks, and Kd are ignored. The diffusive texture (map_Kd) is here used as base color map, since it is typically the same in the old and new parameter definition. The metallic value is between 0 and 1. 1 means VERY reflective, 0 not reflective. People say, you should use either a very high value 0.9-1 or a very small value 0-0.1.

The roughness parameter is also in the range 0-1 and describes the roughness of the material. This should be the most important parameter in the metallic workflow. The metallic workflow is used when one or both of the metallic and roughness values are greater than 0.

It is also possible to use textures which define the metallic and the roughness value dependent on a texture coordinate. For this, use

- map_Km filename

- map_Kr filename

- map_Kmr filename

Km and Kr define a metallic or roughness texture which has three channels, but is used as black and white picture. The parameters map_Kmr defines a texture which implements both, the metallic and the roughness value. Green defines the roughness value and blue defines the metallic value.

5.2 Dielectric workflow

The basic dielectric workflow uses these parameters in the material description:

- Kb r g b (a)

- map_Kd filename

- glossiness x

- specular r g b

Kb defines the base color. The alpha value is optional. Ka, Ks, and Kd are ignored. The diffusive texture (map_Kd) is here used as base color map, since it is typically the same in the old and new parameter definition. The glossiness value is between 0 and 1. 1 means VERY glossy. Glossiness van be seen to be 1-roughness (from the metallic workflow. In addition, you can define the color for specular reflections.

It is also possible to use textures which define the glossiness and the specular color value dependent on a texture coordinate. For this, use

- map_Kg filename

- map_Ks filename

- map_Ksg filename

The combined specular/glossiness texture uses the rgb values for the specular color

and the alpha channel as glossiness value.

5.3 Transparency

It is also possible to use transparency. For this, use an alpha channel in the base texture (map_Kd) and/or define the alpha value as fourth number for the base color, i.e. Kb r g b a. For transparency, set the illum value to 4, as in the usual wavefront parameters.

5.4 Emissive

It is also possible to define emissive colors. For this, you can use the normal wavefront parameters: Ke and map_Ke

5.5 Examples

glossiness workflow

```
newmtl MossieTex
Ns 99.999988
Kb 0.4 0.4 0.4
Ka 1.000000 1.000000 1.000000
Kd 0.800000 0.800000 0.800000
Ks 1 1 1
Ni 1.450000
d 1.000000
illum 2
glossiness 0.8
specular 0.2 0.2 0.2
map_Bump MossieNormal.png
map_Kd MossieTex01.png
```

glossiness workflow, transparent

http://www.heli-x.net

```
newmtl Window
Ns 900.000000
Ka 0.200000 0.200000 0.2
Kd 0.239553 0.239553 0.239553
Ks 0.500000 0.500000 0.500000
Ni 1.000000
d 0.600000
illum 4
Kb 0.5 0.5 0.5 0.6
glossiness 1
specular 0.6 0.6 0.6
```

metallic workflow

```
newmtl chopper_sp
Ns 440.000000
Kb 1 1 1
Ka 0.6 0.6 0.6
Kd 1 1 1
Ks 0.98 0.96 1
Ni 1.450000
d 1.000000
illum 2
metallic 1
roughness 0.1
map_Bump chopper_sp_n2.jpg
map_Kd blade.jpg
```

metallic workflow, transparent   Since the diffusive map contains an alpha channel, three numbers are sufficient for Kb.

```
newmtl top
Ns 160.000013
Ka 0.700000 0.700000 0.700000
Kd 0.400000 0.400000 0.400000
Ks 1.000000 1.000000 1.000000
```

http://www.heli-x.net

```
Ni 1.500000
d 0.500000
illum 4
Kb 1 1 1
metallic 1
roughness 0.3
map_Kd top.png
```

6 Developer

This section is in the state of version 8. We will work to improve the xml file format in the next versions. If you have questions, feel free to contact the develpers dirctly.

6.1 Helicopter Models

Currently, all helicopters are seen as helicopters which have paddles. If you want to model a helicopter without paddles, just set the coupling between paddles and blades to zero.

6.1.1 General remarks

- Models for helicopters consist out of a couple of files which are collected in one directory.

- The name of the helicopter is the name of the directory.

- In order to ensure compatibility with the different operating systems, you have to take care of the cases in the filenames.

- As said before, all files are collected in one directory.

- Three files have the same name as the directory: *.wav with the sound of the helicopter, *.xml with the parameters and *.obj which is a wavefront graphics file.

- There might be other files in the directory, e.g. for the materials or the textures.

- If there is a picture "preview.jpg" in the directory, this picture is used as preview in the model selection window.

- The .obj-File is a wavefront file. Please take care about the following:

  - The center of gravity of the model is the origin of a right handed coordinate system. The x axis points to the front, the y axis is vertical. The z axis points to the right.

      – The rotating axis of the main rotor lies on the y axis.

      – The groups which build the main rotor have names beginning with "Mainrotor".

      – The groups which build the tail rotor have names beginning with "Tailrotor".

      – Groups with transparent material must have a name that ends with `_transparent`, e.g. `canopy_transparent`.

- HELI-X uses different graphical objects for the rotor with low speed and the rotor with high speed. The low speed rotors are the ones from the object file. For rotors with a high speed, a different object is used. If you want specific objects for your helicopter, these objects have to be in the directory with the names "Mainrotor.obj" and "Tailrotor.obj". If these files are not present, HELI-X uses standard rotors.

- Within the xml file there are some parameters for the graphics. They are used for the position of the rotors and for the size of the helicopter.

6.1.2 Visual parameters

In the xml file of the model the following parameters are used for the visual representation of the model.

`<MainrotorPosition>` `[f,f,f]`
    The position of the main rotor in local model coordinates. It is used when the rotor is substituted by the transparent disk.

`<MainrotorTurnAxis>` `[f,f,f]`
    The turn axis of the main rotor.

`<MainrotorAngle>` `[f,f,f]`
    Tipping angles of the axis of the main rotor.

`<MainrotorCrashSizePercentage>` `[f]`
    Using this value (0-100) you can reduce the size of the main rotor with respect to crashes.

`<TailrotorPosition>` `[f,f,f]`

The position of the tail rotor in local model coordinates. It is used when the rotor is substituted by the transparent disk.

`<TailrotorTurnAxis>` [f,f,f]
    The turn axis of the tail rotor.

`<TailrotorAngle>` [f,f,f]
    Tipping angles of the axis of the tail rotor.

`<TailrotorCrashSizePercentage>` [f]
    Using this value (0-100) you can reduce the sensitivity of the tail rotor with respect to crashes.

`<Scale>` [f]
    This parameter can be used to define the scale between model coordinates and real coordinates. Usually, it is not used. Then the actual size of the helicopter is determined by the next two parameters.

`<ScaleDeterminantPartName>` [s]
    Here you write e.g. Mainrotor

`<SizeOfScaleDeterminantPart>` [f]
    and then you specify the diameter of the main rotor here in meter.

`<HelicopterType>` [s]
    Type of the helicopter: Trainer or Scale or Multicopter or Coaxial

`<HookPosition>` [s,f,f,f]
    Defines a hook for the cargo training. A comma seperated list. Name and x,y z coordinates.

`<FactorVisualElasticBladePitch>` [f]
    If you want that a part of the main rotor object deforms due to the pitch or cyclic values, give those parts a group name that contains the string `elastic`. Similarly if you want that the face normals change their angle due to the pitch value, give the corresponding group a name that contains `dynamic_normal`. This factor now determines the amount of elastic bending due to pitch.

`<FactorVisualElasticBladeCyclic>` [f]
    This factor determines the amount of elastic bending of the main rotor due to cyclic controls.

http://www.heli-x.net

`<ColorShape>` [s]

It is possible to change the colors of Shapes within HELI-X. You define the shapes for which this is possible as ColorShape. To identify which color is possible for which shape, there are used groups. An example:

`<ColorShape group="0" color="1">Body</ColorShape>`

There is a shape with the name Body. It is associated with the group 0. The optional value for `color` means, that color 1 is used for this shape.

`<Color>` ,[s,f,f,f,f,f,ff,f,f]

With this tag you define the colors. It is a comma separated list. After the name of the color there are nine numbers between 0 and 1. They define the RGB values of the ambient, diffusive and specular intensities.

`<ColorShapes>` [s,s,...]

Like `ColorShape`, but you can specify several names.

`<TextureShape>` [s]

It is possible to change the texture of the so called texture shapes which are defined with this tag. To identify which texture is possible for which shape, there are used groups. An example:

`<TextureShape group="0" texture="1">Body</TextureShape>`

There is a shape with the name Body. It is associated with the group 0. The optional value for `texture` means, that texture 1 is used for this shape.

`<Texture>` [s,s]

This tag defines a texture. in a comma separated list there is the name and then the file name of the texture. You should also use the attribute group.

`<TextureShapes>` [s,s,...]

Like `TextureShape`, but you can specify several names.

`<GearShape>`   A gear shape is a shape which rotates with the mainrotor. You have to define the `<Name>`, the `<RotationAxis>` (x,y,z), the pivot point `<Pivot>` (x,y,z), and the `<SpeedFactor>`, which defines the relative speed to the main rotor and the direction.

`<CameraPosition>` [f,f,f]

Position of the camera for FPV pilot's view.

`<CameraAngle>` [f]

Angle (degrees) for the camera.

6.1.3 Physics parameters for Helicopters

General parameters

`<Name>` [s]
   The name of the model. It is displayed in the title bar of the window.

`<TimeFactor>` [f]
   This factors determines the simulation speed. Default is 1.

`<PhysicalModelName>` [s]
   The internal name of the physical model. Here you have to write one of these:
   `Helicopter_V1_1`, `CoaxHelicopter_V1_1`, `QuadroCopter_V1_1`, `HexaCopter_V1_1`,
   `TriCopter_V1_1`.

`<Author>` [s]
   The name of the author of the model.

`<WWW>` [s]
   URL of the author's web site.

`<Comment>` [s]
   Comments about the model.

`<Version>` [s]
   The release date of the model in the form YYYY-MM-DD.

`<HelixRevision>` [i]
   The revision number of HELI-X for the model to work.

Mass, geometry, etc.

`<RhoAir>` [f]
   The density of the air in kg/m^3. Default value is 1.209.

`<Mass>` [f]
   The mass of the helicopter in kg.

`<Ixx>` [f]

Moment of inertia for rotation around the x axis. In kg m^2.

`<Iyy>` [f]
Moment of inertia for rotation around the y axis. In kg m^2.

`<Izz>` [f]
Moment of inertia for rotation around the z axis. In kg m^2.

`<Ixy>` [f]
Combined moment of inertia for rotation around the x-y axis. In kg m^2.

Main rotor

`<RotorHeadPosition>` [f,f,f]
The position of the main rotor in the true coordinate system. x,y,z in m.

`<NumberOfBlades>` [i]
The number of blades.

`<MaxRadiusBlade>` [f]
Outer radius of the blades in m.

`<MinRadiusBlade>` [f]
Inner radius of the blades in m.

`<ChordLengthBlade>` [f]
Chord length in m.

`<MassBlade>` [f]
Mass of a blade in kg.

`<SenseOfOrientation>` [i]
Direction of rotation. To the right: -1, to the left: 1.

`<MultiplicatorMainRotorInertia>` [f]
With this factor you increase the moment of inertia of the main rotor around the y axis, e.g. to take into account the paddles.

Main rotor — settings

`<PitchAngleMinMax>` [f,f]
  Minimum and maximum pitch angle in degree.

`<BladeRollAngleMinMax>` [f,f]
  Minimum and maximum roll angle in degree.

`<BladeNickAngleMinMax>` [f,f]
  Minimum and maximum nick angle in degree.

`<RotorRevPerMin100>` [f]
  Number of revolutions per minute of the main rotor. The value gives the speed
  when the throttle curve has the value +100%.

`<PitchCurve>` [i,i,i,i,i]
  5 values, which define the pitch curve. Values from -100 to 100.

`<PitchCurve2>` [i,i,i,i,i]
  Like `<PitchCurve>`, but for Idle Up 2.

`<PitchCurveA>` [i,i,i,i,i]
  Like `<PitchCurve>`, but for auto rotation.

`<ThrottleCurve>` [i,i,i,i,i]
  5 values, which define the throttle curve. Values from -100 to 100.

`<ThrottleCurve2>` [i,i,i,i,i]
  Like `<ThrottleCurve>`, but for Idle Up 2.

`<ThrottleCurveA>` [i,i,i,i,i]
  Like `<ThrottleCurve>`, but for auto rotation.

`<NickExpo>` [i]
  Expo settings for cyclic nick, -100 up to 100. Default is 0.

`<NickExpo2>` [i]
  Like `<NickExpo>`, but for Idle Up 2.

`<NickExpoA>` [i]
  Like `<NickExpo>`, but for auto rotation.

http://www.heli-x.net

`<NickDualRate>` [i]
Dual rate settings for cyclic nick, 25 up to 125. Default is 100.

`<NickDualRate2>` [i]
Like `<NickDualRate>`, but for Idle Up 2.

`<NickDualRateA>` [i]
Like `<NickDualRate>`, but for auto rotation.

`<RollExpo>` [i]
Expo settings for cyclic nick, -100 up to 100. Default is 0.

`<RollExpo2>` [i]
Like `<RollExpo>`, but for Idle Up 2.

`<RollExpoA>` [i]
Like `<RollExpo>`, but for auto rotation.

`<RollDualRate>` [i]
Dual rate settings for cyclic roll, 25 up to 125. Default is 100.

`<RollDualRate2>` [i]
Like `<RollDualRate>`, but for Idle Up 2.

`<RollDualRateA>` [i]
Like `<RollDualRate>`, but for auto rotation.

Main rotor — thrust

`<MultiplicatorRotorLift>` [f]
This factor is used to fit the thrust of the rotor.

`<MultiplicatorRotorLiftAutorotation>` [f]
Same as `MultiplicatorRotorLift`, but valid during auto rotation.

`<MultiplicatorAirResistanceM>` [f]
This factor determines the air resistance of the rotor which is independent of the angle of attack.

`<MultiplicatorAirResistanceN>` [f]
    This factor determines the air resistance of the rotor which is dependent on the angle of attack.

`<MultiplicatorStallLift>` [f]
    This factor controls the lift during stall.

`<FactorAirResistanceDuringStall>` [f]
    During stall you can influence the air resistance (dependent on pitch) with this factor.

`<StallAngle>` [f]
    This is the angle (in degree), for which a stall might occur.

`<StallAngleAutorotation>` [f]
    Like StallAngle, but valid during the autorotation phase.

`<StallRevPerMin>` [f]
    The stall can only happen when the number of revolutions is below this number.

`<MultiplicatorNickRollPower>` [f]
    This factor increases the power consumption caused by cyclic controls. It is due to the air resistance of the paddles.

`<AdditionalLinearAirResistance>` [f]
    With this parameter you can adjust the air resistance during forward flight (independent of the angle of attack).

`<AdditionalLinearPitchAirResistance>` [f]
    With this parameter you can adjust the air resistance during forward flight (dependent on the angle of attack).

`<GroundEffectExponent>` [f]
    Tuning of the ground effect. Default is 1. Higher values will increase the ground effect.

`<GroundEffectIntensity>`[f]
    Tuning of the ground effect. Default is 1. Higher values will increase the horizontal instability.

`<TipLossFactor>` [f]

The thrust at the tips of the blades is not optimal. If you decrease the value (default is 1) you can decrease the thrust in order to simulate the tip losses.

`<VerticalClimbFactor>` [f]
This factor is very experimental. Values greater than the default value 1 will restrict the maximal climb speed of the helicopter. Maybe this parameter will be erased at some time.

`<MultiplicatorRotorTorque>` [f]
Factor to increase the torque caused by the main rotor.

`<EulerFlowMagicNumber1>` [f]
This parameters controls the inertia of the air flowing through the main rotor. This influences the reaction speed in pitch changes.

`<EulerFlowMagicNumber2>` [f]
This factor controls the flow situation in forward flight. It is responsible for the gain of lift during forward flight.

`<EulerFlowMagicNumber3>` [f]
This parameter controls the shape of the air flow above and below the main rotor. This controls e.g. the maximum climb velocity.

`<EulerFlowMagicNumber4>` [f]
The parameter controls the lift in forward flight.

`<FactorVortexIntensity>` [f]
Strength of the vortex ring state.

`<FactorVortexDiffusion>` [f]
Diffusion constant of the vortex ring state.

`<FactorRollNickCouplingForewardFlight>` [f]
During forward flight torque is also produced in roll direction. This can be influenced by this value.

`<PitchDynamic>` [f]
Shortens the reaction time to pitch which is due to the dynamics of the blade angles. It makes the heli more aggressive, so use it with care. Default: 0:

`<NickDynamic>` [f]

Shortens the reaction time to nick which is due to the dynamics of the blade angles. It makes the heli more aggressive, so use it with care. Default: 0:

`<RollDynamic>` [f]

Shortens the reaction time to roll which is due to the dynamics of the blade angles. It makes the heli more aggressive, so use it with care. Default: 0:

Main rotor — rotor head

`<RotorHeadStiffnessTimeConstant>` [f]

The bearing of the feathering shaft can cause vibrations. This is the time constant of theses vibrations (in seconds).

`<RotorHeadStiffnessDamping>` [f]

The damping of the system is determined by this parameter. Higher values mean a greater damping. Small values cause intensive vibrations. Tha value 1 is the limit between oscillating systems and strongly damped systems.

`<RotorHeadStiffnessMinimalDamping>` [f]

For small rotation speeds the vibrations are typically more intensive. Thus, this damping value is used for small rotation speeds.

`<RotorHeadStiffnessMinimalOmega>` [f]

And this damping value is used for revolutions per minute which are smaller than this value.

`<RotorHeadStiffnessNickRollCoupling>` [f]

The coupling of the vibrations in nick and roll direction is determined by this value. Zero means no coupling, larger values mean a greater coupling.

`<RotorHeadInstabilityTimeConstant>` [f]

Time constant (in seconds) for the rotor head instability. A good starting point is 2.

`<RotorHeadStiffnessDifferential>` [f]

This value (when greater than the default value 0) causes an amplification of the vibrations based on the vibration velocity.

`<RotorHeadInstabilityAmplitude>` [f]

Amplitude of the rotor head instability.

`<RotorPhaseShiftDegrees>` [f]
The pase shift (in) between controlled and true cyclic controls.

Bell–Hiller paddles

`<MassPaddle>` [f]
The mass of a paddle in kg.

`<MassFlybar>` [f]
The mass of the flybar in kg.

`<LengthFlybar>` [f]
The length of the flybar in m.

`<AreaPaddle>` [f]
Area of one paddle in m^2.

Bell–Hiller paddles — settings

`<PaddlePitchFactor>` [f]
Factor between angles of the flybar and the angle of attack of the blades. Example: 1 means the angles of the flybar is equal to the angle of the blades. 0 is equivalent to a rotor without paddles.

`<HillerNickAngleMinMax>` [f]
Minimum and maximum nick angle of the paddles in degree. If there are two values, the first is for idle up 1, the second for idle up 2.

`<HillerRollAngleMinMax>` [f]
Minimum and maximum roll angle of the paddles in degree. If there are two values, the first is for idle up 1, the second for idle up 2.

Tail rotor

`<TailNumberOfBlades>` [i]

http://www.heli-x.net

Number of blades of the tail rotor.

`<TailMaxRadiusBlade>` [f]
    Outer radius of the tail rotor in m.

`<TailMinRadiusBlade>` [f]
    Inner radius of the tail blades in m.

`<TailChordLengthBlade>` [f]
    Chord length of the tail blades in m.

`<TailXyzPosition>` [f,f,f]
    Position of the tail rotor, x,y,z, in m.

`<TailXyzDirection>` [f,f,f]
    Direction of the rotation axis of the tail rotor.

`<TailFrequencyRatio>` [f]
    Frequency ratio between the revolutions of the tail rotor and the main rotor.

Tail rotor — thrust

`<MultiplicatorTailRotorLift>` [f]
    This factor is used to fit the thrust of the tail rotor.

`<MultiplicatorTailAirResistanceM>` [f]
    This factor determines the air resistance of the tail rotor which is independent of the angle of attack.

`<MultiplicatorTailAirResistanceN>` [f]
    This factor determines the air resistance of the tail rotor which is dependent on the angle of attack.

`<MultiplicatorSideShiftByTailRotor>` [f]
    This factor increases the sideward thrust of the tail rotor. Default is 1.

`<TailRotorFlightAirResistanceM>` [f]
    This factor determines the air resistance of the tail rotor in forward flight which is independent of the angle of attack.

`<TailRotorFlightAirResistanceN>` [f]
    This factor determines the air resistance of the tail rotor in forward flight which is dependent on the angle of attack.

Tail rotor, gyro — settings

`<TailPitchAngleMinMax>` [f,f]
    Smallest and greatest angle of attack of the tail blades.

`<TailServoSpeed>` [f]
    Maximum speed of the tail servo in pitch degrees per second.

`<TailThrustReduction>` [f]
    This exponent is responsible for a non proportional reduction of the thrust. Default: 1. Values greater than 1 will make a stall at small numbers of revolution more likely.

`<YawRateMax>` [f]
    Maximum rotation speed around the vertical axis in revolutions per second.

`<GyroGain>` [f]
    Gyro gain.

`<GyroGain2>` [f]
    Gyro gain for Idle Up 2.

`<GyroGainA>` [f]
    Gyro gain during auto rotation.

`<GyroHeadingHoldGain>` [f]
    Gain value for heading hold. 0 means no heading hold.

`<GyroHeadingHoldGain2>` [f]
    Like `<GyroHeadingHoldGain>`, but for Idle Up 2.

`<GyroHeadingHoldGainA>` [f]
    Like `<GyroHeadingHoldGain>`, but for Autorotation.

`<GyroSensorQuality>` [f]

For the simulation of the gyro drift. Value 0: no drift. Value greater than 0: Yes there is a drift.

`<YawExpo>` [i]
Expo setting for yaw, -100 up to 100. Default is 0.

`<YawExpo2>` [i]
Like `<YawExpo>`, but for Idle Up 2.

`<YawExpoA>` [i]
Like `<YawExpo>`, but for Autorotation.

`<YawDualRate>` [i]
Dual rate setting for yaw, 25 up to 125. Default is 100.

`<YawDualRate2>` [i]
Like `<YawDualRate>`, but for Idle Up 2.

`<YawDualRateA>` [i]
Like `<YawDualRate>`, but for Autorotation.

3-Axes-Stabilisation system

`<StabiNickGain>` [f]
P-Factor for the control of cyclic nick.

`<StabiNickHeadingHoldGain>` [f]
I-factor for the control of cyclic nick.

`<StabiNickSensorQuality>` [f]
Quality of the sensor. 0 means best quality.

`<StabiNickRateMax>` [f]
The maximum rate in revolutions per second.

`<StabiNickFeedForwardFactor>` [f]
The feed forward factor for nick, default value is zero, maximal value is 1.

`<StabiRollGain>` [f]

P-Factor for the control of cyclic roll.

`<StabiRollHeadingHoldGain>` [f]
I-factor for the control of cyclic roll.

`<StabiRollSensorQuality>` [f]
Quality of the sensor. 0 means best quality.

`<StabiRollRateMax>` [f]
The maximum rate in revolutions per second.

`<StabiRollFeedForwardFactor>` [f]
The feed forward factor for roll, default value is zero, maximal value is 1.

`<VStabiPitchFactor>` [f]
The corresponding value of the V-Stabi system.

`<VStabiExpoModeYaw>` [i]
The corresponding value of the V-Stabi system.

`<VStabiExpoModeSwashplate>` [i]
The corresponding value of the V-Stabi system.

Engine

`<EngineFrequencyRatio>` [f]
Frequency ratio of the engine to the main rotor.

`<OmegaTorqueData>` [f,f,...]
Table with the engine characteristics. Each pair represents a number of revolution per minute and the torque in Nm.

`<GearFrictionCoefficientNormal>` [f]
Friction coefficient for the friction which is proportional to the speed.

`<GearFrictionCoefficientAutorotation>` [f]
Same as above for auto rotation. Usually smaller than the normal value.

Servos

`<ServoMaxSpeed>` [f]
> Maximum angular speed of a servo in degree per second.

`<ServoClearance>` [f]
> Clearance of the servos in degree.

Fuselage

`<Face>` [node]
> The fuselage is approximated using faces. Each face is represented by a list of the following tags:

`<Position>` [f,f,f]
> The coordinates of the midpoint of the surface.

`<Normal>` [f,f,f]
> The direction of the normal vector (pointing outwards) of the face.

`<CW>` [f]
> The C_W-value if the face.

`<Area>` [f]
> the area of the face in m^2.

`<MultiplicatorAirResistanceTailRotorBroken>` [f]
> In case of a tail rotor crash, this parameter can be used to increase the air resistance against the rotation of the fuselage.

Horizontal and vertical tails

`<VerticalTailArea>` [f]
> Area of the vertical tail plane.

`<VerticalTailNormal>` [f,f,f]
> A vector which is normal to the vertical tail plane. Often, 0,0,1 is correct.

http://www.heli-x.net

`<VerticalTailPosition>` [f,f,f]
    The position of the midpoint of the vertical tail plane.

`<HorizontalTailArea>` [f]
    Area of the horizontal tail plane.

`<HorizontalTailNormal>` [f,f,f]
    A vector which is normal to the horizontal tail plane. Often, 0,1,0 is correct.

`<HorizontalTailPosition>` [f,f,f]
    The position of the midpoint of the horizontal tail plane.

Landing gear

`<ContactStiffness>` [f]
    Stiffness of the landing gear.

`<ContactViscosity>` [f]
    Damping of the landing gear in normal direction.

`<ContactViscosityTangential>` [f]
    Damping of the landing gear in tangential direction.

`<FrictionCoefficientX>` [f]
    Friction coefficient in x-direction, i.e. forwards.

`<FrictionCoefficientZ>` [f]
    Friction coefficient in z-direction, i.e. sidewards.

`<MaximalContactVelocity>` [f]
    The maximal velocity during contact so that there is no crash (in m/s). Default: 5.

Sound

`<SoundVolumeFactor>` [f]
    Loudness factor of the engine sound, between 0 and 1.

`<BladeVolumeFactor>` [f]

Loudness factor of the blade sound, between 0 and 1.

`<TailVolumeFactor> [f]`
Loudness factor of the tail sound, between 0 and 1.

`<CrashVolumeFactor> [f]`
Loudness factor of the crash sound, between 0 and 1.

`<SoundPitchFactor> [f]`
A factor which scales the influence of the pitch of the rotor. Default value is 1.

`<SoundMinPitch> [f]`
The influence of the pitch value is restricted. The minimum pitch is set by this factor. Default value is 0.5.

`<SoundMaxPitch> [f]`
The influence of the pitch value is restricted. The maximum pitch is set by this factor. Default value is 1.5.

`<BladePitchFactor> [f]`
Similar as `SoundPitchFactor` for the blade sound.

`<BladeMinPitch> [f]`
Similar as `SoundMinPitch` for the blade sound.

`<BladeMaxPitch> [f]`
Similar as `SoundMaxPitch` for the blade sound.

`<TailPitchFactor> [f]`
Similar as `SoundPitchFactor` for the tail sound.

`<TailMinPitch> [f]`
Similar as `SoundMinPitch` for the tail sound.

`<TailMaxPitch> [f]`
Similar as `SoundMaxPitch` for the tail sound.

`<SoundConeExponentMain> [f]`
Determines how sharp the directional cone for the blade sound is. 0 means: no directional dependence, 10 means very strnng directional dependence. Default value: 1.

`<SoundDirectionDependenceFactorMain>` [f]
: Determines how much of the complete blade sound is directional dependent. 1 means: The whole sound is directional dependent, 0 means: no directional dependence. Default value: 0.7

`<SoundConeExponentTail>` [f]
: Determines how sharp the directional cone for the tail sound is. 0 means: no directional dependence, 10 means very strnng directional dependence. Default value: 1.

`<SoundDirectionDependenceFactorTail>` [f]
: Determines how much of the whole tail sound is directional dependent. 1 means: The complete sound is directional dependent, 0 means: no directional dependence. Default value: 0.7

Smoke

`<SmokeIntensityFactor>` [f]
: The intensity of the smoke. Usually values in the range 0.5 up to 5.

`<SmokeDurationSeconds>` [f]
: The life time of a smoke particle in seconds.

`<SmokeDiffusionConstant>` [f]
: The value defines how fast the smoke spreads out in the air.

`<SmokeTimeConstant>` [f]
: The time constant determined how fast the smoke follows the speed of the surrounding air. Values which are too small (less than 0.01) might cause instabilities in the smoke simulation.

`<SmokeColorRGB>` [f,f,f]
: The smoke color as RGB value.

`<SmokeSourcePosition>` [f,f,f]
: The position of the smoke source in model coordinates.

`<SmokeDirection>` [f,f,f]
: The initial direction of the smoke velocity.

`<SmokeEmissionVelocity>` [f]
> The speed of the smoke in m/s.

`<SmokeEmissionSpread>` [f]
> This value determined the diameter of the cone of the smoke generation. Typical values are in the range 0.1 to 1.

`<SmokeBlendingPreset1to8>` [i]
> Here you define one out of 8 presets how the blending is realized in terms of openGL.

`<SmokeTextureNumber>` [i]
> You choose the texture of the smoke with an integer value between 0 and 5. Default value: 0.

`<Smoke2IntensityFactor>` [f]
> The intensity of the smoke. Usually values in the range 0.5 up to 5. For the second smoke source.

`<Smoke2DurationSeconds>` [f]
> The life time of a smoke particle in seconds. For the second smoke source.

`<Smoke2DiffusionConstant>` [f]
> The value defines how fast the smoke spreads out in the air. For the second smoke source.

`<Smoke2TimeConstant>` [f]
> The time constant determined how fast the smoke follows the speed of the surrounding air. Values which are too small (less than 0.01) might cause instabilities in the smoke simulation. For the second smoke source.

`<Smoke2ColorRGB>` [f,f,f]
> The smoke color as RGB value. For the second smoke source.

`<Smoke2SourcePosition>` [f,f,f]
> The position of the smoke source in model coordinates. For the second smoke source.

`<Smoke2Direction>` [f,f,f]
> The initial direction of the smoke velocity. For the second smoke source.

`<Smoke2EmissionVelocity>` [f]
> The speed of the smoke in m/s. For the second smoke source.

`<Smoke2EmissionSpread>` [f]
> This value determined the diameter of the cone of the smoke generation. Typical values are in the range 0.1 to 1. For the second smoke source.

`<Smoke2BlendingPreset1to8>` [i]
> Here you define one out of 8 presets how the blending is realized in terms of openGL. For the second smoke source.

`<Smoke2TextureNumber>` [i]
> You choose the texture of the smoke with an integer value between 0 and 5. Default value: 0. For the second smoke source.

6.1.4 Physics parameters for Planes

General parameters

`<Name>` [s]
> The name of the model. It is displayed in the title bar of the window.

`<TimeFactor>` [f]
> This factors determines the simulation speed. Default is 1.

`<PhysicalModelName>` [s]
> The internal name of the physical model. Here you have to write one of these: `PlaneMonoStandard`.

`<Author>` [s]
> The name of the author of the model.

`<WWW>` [s]
> URL of the author's web site.

`<Comment>` [s]
> Comments about the model.

`<Version>` [s]

The release date of the model in the form YYYY-MM-DD.

```
<HelixRevision> [i]
```
The revision number of HELI-X for the model to work.

Visual Parameters

```
<VisualPropellerPosition> [f,f,f]
```

```
<VisualPropellerTurnAxis> [f,f,f]
```

```
<VisualPropellerPlateAngle> [f,f,f]
```

```
<VisualLeftAileronMinMaxAngle> [f,f]
```

```
<VisualLeftAileronPivot> [f,f,f]
```

```
<VisualLeftAileronSecondPoint> [f,f,f]
```

```
<VisualRightAileronMinMaxAngle> [f,f]
```

```
<VisualRightAileronPivot> [f,f,f]
```

```
<VisualRightAileronSecondPoint> [f,f,f]
```

```
<VisualLeftLandingFlapMinMaxAngle> [f,f]
```

```
<VisualLeftLandingFlapPivot> [f,f,f]
```

`<VisualLeftLandingFlapSecondPoint>` [f,f,f]

`<VisualLeftLandingFlapMaxSpeed>` [f]

`<VisualRightLandingFlapMinMaxAngle>` [f,f]

`<VisualRightLandingFlapPivot>` [f,f,f]

`<VisualRightLandingFlapSecondPoint>` [f,f,f]

`<VisualRightLandingFlapMaxSpeed>` [f]

`<VisualRudderMinMaxAngle>` [f,f]

`<VisualRudderPivot>` [f,f,f]

`<VisualRudderSecondPoint>` [f,f,f]

`<VisualNoseWheelMinMaxAngle>` [f,f]

`<VisualNoseWheelPivot>` [f,f,f]

`<VisualNoseWheelSecondPoint>` [f,f,f]

`<VisualLeftElevatorMinMaxAngle>` [f,f]

`<VisualLeftElevatorPivot>` [f,f,f]

`<VisualLeftElevatorSecondPoint>` [f,f,f]

`<VisualRightElevatorMinMaxAngle>` [f,f]

`<VisualRightElevatorPivot>` [f,f,f]

`<VisualRighttElevatorSecondPoint>` [f,f,f]

`<HookPosition>` [s,f,f,f]

`<CameraPosition>` [f,f,f]

`<HandLaunchSpeed>` [f]

`<HandLaunchStandard>` [s]

`<VisualWingSpan>` [s]

`<PlaneType>` [s]

Visual Propeller

`<PropellerRevPerMinMax>` [f]

`<VisualSenseOfOrientation>` [f]

Mass, geometry, etc.

`<RhoAir>` [f]
   The density of the air in kg/m^3. Default value is 1.209.

`<Mass>` [f]
   The mass of the helicopter in kg.

`<Ixx>` [f]
   Moment of inertia for rotation around the x axis. In kg m^2.

`<Iyy>` [f]
   Moment of inertia for rotation around the y axis. In kg m^2.

`<Izz>` [f]
   Moment of inertia for rotation around the z axis. In kg m^2.

`<IxyAngleDeg>` [f]
   Degrees of the angle between the x axis and the main axis of intertia.

Profile

`<ProfileType>` [s]
   The following types are supported: `Standard`, `Pendulum`, `Airbrake`, `SymmetricPendulum`.

Profile Standard

`<FlapsMaxAngle>` [f]

`<ClMax>` [f]

`<AngleClMax>` [f]

`<FlapsClMax>` [f]

`<FlapsAngleClMax>` [f]

`<AngleDuringFullLift>` [f]

`<AngleAfterStall>` [f]

`<FactorLiftAfterStall>` [f]

`<ReynoldsAngle>` [f]

`<ReferenceReynoldsNumber>` [f]

`<FactorBackwardsLift>` [f]

`<Cd0>` [f]

`<CdMax>` [f]

`<FactorCm>` [f]

`<FactorReynoldsInflucenceCd>` [f]

`<FactorFlapInfluenceCD>` [f]

`<SymmetryPercentage>` [f]

Profile Pendulum

`<ClMax>` [f]

`<AngleClMax>` [f]

`<AngleDuringFullLift>` [f]

`<AngleAfterStall>` [f]

`<FactorLiftAfterStall>` [f]

`<ReynoldsAngle>` [f]

`<ReferenceReynoldsNumber>` [f]

`<FactorBackwardsLift>` [f]

`<Cd0>` [f]

`<CdMax>` [f]

`<FactorCm>` [f]

`<FactorReynoldsInflucenceCd>` [f]

`<SymmetryPercentage>` [f]

Profile Symmetric Pendulum

`<ClMax>` [f]

`<AngleClMax>` [f]

`<ClMaxSharpnessPercent>` [f]

`<SecondPointCl>` [f]

`<SecondPointAngleCl>` [f]

`<Cd0>` [f]

`<CdMax>` [f]

`<CdMaxLift>` [f]

`<FactorCm>` [f]

Profile Airbrake

`<FlapsMaxAngle>` [f]

`<ClMax>` [f]

`<AngleClMax>` [f]

http://www.heli-x.net

`<AngleDuringFullLift>` [f]

`<AngleAfterStall>` [f]

`<FactorLiftAfterStall>` [f]

`<ReynoldsAngle>` [f]

`<ReferenceReynoldsNumber>` [f]

`<FactorBackwardsLift>` [f]

`<Cd0>` [f]

`<CdMax>` [f]

`<FactorCm>` [f]

`<FactorReynoldsInflucenceCd>` [f]

`<AdditionalFlapCD>` [f]

`<SymmetryPercentage>` [f]

Wing

`<WingType>` [s]

http://www.heli-x.net

Currently only `Rectangular`

`<Position>` [f,f,f]

`<InnerWingSpan>` [f]

`<WingSpan>` [f]

`<ChordLength>` [f]

`<DihedralAngle>` [f]

`<IncidenceAngle>` [f]

`<MaxFlapAngle>` [f]

`<MaxElevonAngle>` [f]

`<PropellerReactionPercentage>` [f]

`<FactorPropellerReductionSpin>` [f]

`<PropellerSwirlReactionPercentage>` [f]

`<FactorLocalFlowDuringPropellerInfluencePercentage>` [f]

`<FactorInducedDrag>` [f]

`<FactorPropellerReductionSpin>` [f]

`<ReynoldsPercentageAngle>` [f]

`<ReynoldsPercentageLift>` [f]

`<InfluenceMainWingPercentage>` [f]

`<TimeConstantInfluenceMainWing>` [f]

`<LandingFlapAngle>` [f,]

`<AirbrakeAngle>` [f,]

HorizontalStabilizer

`<WingType>` [s]
 Currently only `Rectangular`

`<Position>` [f,f,f]

`<InnerWingSpan>` [f]

`<WingSpan>` [f]

`<ChordLength>` [f]

`<DihedralAngle>` [f]

`<IncidenceAngle>` [f]

`<MaxFlapAngle>` [f]

`<PropellerReactionPercentage>` [f]

`<FactorPropellerReductionSpin>` [f]

`<PropellerSwirlReactionPercentage>` [f]

`<FactorLocalFlowDuringPropellerInfluencePercentage>` [f]

`<FactorInducedDrag>` [f]

`<FactorPropellerReductionSpin>` [f]

`<ReynoldsPercentageAngle>` [f]

`<ReynoldsPercentageLift>` [f]

`<InfluenceMainWingPercentage>` [f]

`<TimeConstantInfluenceMainWing>` [f]

VerticalStabilizer

http://www.heli-x.net

`<Position>` [f,f,f]

`<Area>` [f]

`<MaxFlapAngle>` [f]

`<PropellerReactionPercentage>` [f]

`<FactorPropellerReductionSpin>` [f]

`<PropellerSwirlReactionPercentage>` [f]

`<FactorLocalFlowDuringPropellerInfluencePercentage>` [f]

`<FactorInducedDrag>` [f]

`<FactorPropellerReductionSpin>` [f]

`<ReynoldsPercentageAngle>` [f]

`<ReynoldsPercentageLift>` [f]

Fuselage

`<Position>` [f,f,f]

`<Length>` [f]

```
<Length> [f]
```

```
<Height> [f]
```

RetractableLandingGear

```
<Position> [f,f,f]
```

```
<Linear> [f]
```

```
<Quadratic> [f]
```

RotationalDamping

```
<LinearX> [f]
```

```
<QuadraticX> [f]
```

```
<LinearY> [f]
```

```
<QuadraticY> [f]
```

```
<LinearZ> [f]
```

```
<QuadraticZ> [f]
```

VelocityDamping

`<LinearX>` [f]

`<QuadraticX>` [f]

`<LinearY>` [f]

`<QuadraticY>` [f]

`<LinearZ>` [f]

`<QuadraticZ>` [f]

PowerTrain

`<Type>` [s]
The following types are supported: `Jet`, `Nitro`, `Electric`.

PowerTrain Jet

`<PropellerPosition>` [f,f,f]

`<PropellerTurnAxis>` [f,f,f]

`<RevPerMinuteAtMaxThrust>` [f]

`<RevPerMinuteStandard>` [f]

`<RevPerMinuteIdle>` [f]

`<ThrustMax>` [f]

`<ExpoPercent>` [f]

`<TimeConstant>` [f]

`<TimeConstantLowRPM>` [f]

PowerTrain Nitro or Electric

`<PropellerPosition>` [f,f,f]

`<PropellerTurnAxis>` [f,f,f]

`<PropellerDiameter>` [f,f,f]

`<RevPerMinuteAtMaxThrust>` [f]

`<RevPerMinuteStandard>` [f]

`<RevPerMinuteIdle>` [f]

`<ThrustMax>` [f]

`<TorqueFactor>` [f]

`<SenseOfRotation>` [f]

`<FactorThrustReductionSpin>` [f]

`<WindMillFactor>` [f]

`<FoldingPropeller>` [b]

`<StickFriction>` [f]

`<SlipFriction>` [f]

`<ExpoPercent>` [f]

`SoundExpoPercent<>` [f]

`<TimeConstant>` [f]

Landing gear

`<ContactStiffness>` [f]
    Stiffness of the landing gear.

`<ContactViscosity>` [f]
    Damping of the landing gear in normal direction.

`<ContactViscosityTangential>` [f]
    Damping of the landing gear in tangential direction.

`<FrictionCoefficientX>` [f]
    Friction coefficient in x-direction, i.e. forwards.

`<FrictionCoefficientZ>` [f]
    Friction coefficient in z-direction, i.e. sidewards.

`<MaximalContactVelocity>` [f]
    The maximal velocity during contact so that there is no crash (in m/s). Default: 5.

`<WheelContactStiffness>` [f]

`<WheelContactViscosity>` [f]

`<WheelContactViscosityTangential>` [f]

`<WheelFrictionFactorX>` [f]

`<WheelFrictionFactorZ>` [f]

`<NoseWheelContactStiffness>` [f]

`<NoseWheelContactViscosity>` [f]

`<NoseWheelContactViscosityTangential>` [f]

`<NoseWheelFrictionFactorX>` [f]

`<NoseWheelFrictionFactorZ>` [f]

`<TailWheelContactStiffness>` [f]

`<TailWheelContactViscosity> [f]`

`<TailWheelContactViscosityTangential> [f]`

`<TailWheelFrictionFactorX> [f]`

`<TailWheelFrictionFactorZ> [f]`

Sound

`<SoundVolumeFactor> [f]`
   Loudness factor of the engine sound, between 0 and 1.

`<CrashVolumeFactor> [f]`
   Loudness factor of the crash sound, between 0 and 1.

`<SoundPitchFactor> [f]`
   A factor which scales the influence of the pitch of the rotor. Default value is 1.

`<SoundMinPitch> [f]`
   The influence of the pitch value is restricted. The minimum pitch is set by this
   factor. Default value is 0.5.

`<SoundMaxPitch> [f]`
   The influence of the pitch value is restricted. The maximum pitch is set by this
   factor. Default value is 1.5.

`<SoundConeExponentMain> [f]`
   Determines how sharp the directional cone for the blade sound is. 0 means: no
   directional dependence, 10 means very strnng directional dependence. Default
   value: 1.

`<SoundDirectionDependenceFactorMain> [f]`
   Determines how much of the complete blade sound is directional dependent.
   1 means: The whole sound is directional dependent, 0 means: no directional

dependence. Default value: 0.7

`<SoundPropellerMinSpeed>` [f]

`<SoundPropellerMaxSpeed>` [f]

`<SoundPropellerConeExponent>` [f]

`<SoundPropellerDirectionDependenceFactor>` [f]

`<SoundWindVolumeFactor>` [f]

`<SoundWindMinSpeed>` [f]

`<SoundWindMaxSpeed>` [f]

`<SoundWindConeExponent>` [f]

`<SoundWindDirectionDependenceFactor>` [f]

`<RotorRevPerMin100>` [f]

Smoke inside a Block Smoke you can define:

`<SmokeIntensityFactor>` [f]
    The intensity of the smoke. Usually values in the range 0.5 up to 5.

`<SmokeDurationSeconds>` [f]
    The life time of a smoke particle in seconds.

http://www.heli-x.net

`<SmokeDiffusionConstant>` `[f]`
> The value defines how fast the smoke spreads out in the air.

`<SmokeTimeConstant>` `[f]`
> The time constant determined how fast the smoke follows the speed of the surrounding air. Values which are too small (less than 0.01) might cause instabilities in the smoke simulation.

`<SmokeColorRGB>` `[f,f,f]`
> The smoke color as RGB value.

`<SmokeSourcePosition>` `[f,f,f]`
> The position of the smoke source in model coordinates.

`<SmokeDirection>` `[f,f,f]`
> The initial direction of the smoke velocity.

`<SmokeEmissionVelocity>` `[f]`
> The speed of the smoke in m/s.

`<SmokeEmissionSpread>` `[f]`
> This value determined the diameter of the cone of the smoke generation. Typical values are in the range 0.1 to 1.

`<SmokeBlendingPreset1to8>` `[i]`
> Here you define one out of 8 presets how the blending is realized in terms of openGL.

`<SmokeTextureNumber>` `[i]`
> You choose the texture of the smoke with an integer value between 0 and 5. Default value: 0.

6.1.5 Physics Parameters for Multicopters

General parameters

`<Name>` `[s]`
> The name of the model. It is displayed in the title bar of the window.

http://www.heli-x.net

`<TimeFactor>` [f]
  This factors determines the simulation speed. Default is 1.

`<PhysicalModelName>` [s]
  The internal name of the physical model. Here you have to write one of these:
  `Helicopter_V1_1,CoaxHelicopter_V1_1,QuadroCopter_V1_1,HexaCopter_V1_1,`
  `TriCopter_V1_1`.

`<Author>` [s]
  The name of the author of the model.

`<WWW>` [s]
  URL of the author's web site.

`<Comment>` [s]
  Comments about the model.

`<Version>` [s]
  The release date of the model in the form YYYY-MM-DD.

`<HelixRevision>` [i]
  The revision number of HELI-X for the model to work.

Mass, geometry, etc.

`<RhoAir>` [f]
  The density of the air in kg/m^3. Default value is 1.209.

`<Mass>` [f]
  The mass of the helicopter in kg.

`<Ixx>` [f]
  Moment of inertia for rotation around the x axis. In kg m^2.

`<Iyy>` [f]
  Moment of inertia for rotation around the y axis. In kg m^2.

`<Izz>` [f]
  Moment of inertia for rotation around the z axis. In kg m^2.

`<Ixy>` [f]
Combined moment of inertia for rotation around the *x-y* axis. In kg m^2.

`<ConfigurationX>` [b]
true or false, depending on configuration type.

Main rotor

`<RotorHeadPosition>` [f,f,f]
The position of the main rotor in the true coordinate system. x,y,z in m.

`<RotorHeadPosition2>` [f,f,f]
The position of the main rotor 2 in the true coordinate system. x,y,z in m.

`<RotorHeadPosition3>` [f,f,f]
The position of the main rotor 3 in the true coordinate system. x,y,z in m.

`<RotorHeadPosition4>` [f,f,f]
The position of the main rotor 4 in the true coordinate system. x,y,z in m.

`<RotorHeadPosition5>` [f,f,f]
The position of the main rotor 5 in the true coordinate system. x,y,z in m (only for hexacopters).

`<RotorHeadPosition6>` [f,f,f]
The position of the main rotor 6 in the true coordinate system. x,y,z in m (only for hexacopters).

`<NumberOfBlades>` [i]
The number of blades.

`<MaxRadiusBlade>` [f]
Outer radius of the blades in m.

`<MinRadiusBlade>` [f]
Inner radius of the blades in m.

`<ChordLengthBlade>` [f]
Chord length in m.

http://www.heli-x.net

`<MassBlade>` [f]
    Mass of a blade in kg.

`<SenseOfOrientation>` [i]
    Direction of rotation. To the right: -1, to the left: 1.

`<MultiplicatorMainRotorInertia>` [f]
    With this factor you increase the moment of inertia of the main rotor around the
    y axis, e.g. to take into account the paddles.

Main rotor — settings

`<PitchAngleMinMax>` [f,f]
    Minimum and maximum pitch angle in degree.

`<RotorRevPerMin100>` [f]
    Number of revolutions per minute of the main rotor. The value gives the speed
    when the throttle curve has the value +100%.

`<MaxDiffRotorPercentage>` [f]
    The maximum difference of the rotors in percent. Using this parameter you can
    avoid that a rotor is completely turned off. Default: 15.

`<PitchCurve>` [i,i,i,i,i]
    5 values, which define the pitch curve. Values from -100 to 100.

`<PitchCurve2>` [i,i,i,i,i]
    Like `<PitchCurve>`, but for Idle Up 2.

`<PitchCurveA>` [i,i,i,i,i]
    Like `<PitchCurve>`, but for auto rotation.

`<ThrottleCurve>` [i,i,i,i,i]
    5 values, which define the throttle curve. Values from -100 to 100.

`<ThrottleCurve2>` [i,i,i,i,i]
    Like `<ThrottleCurve>`, but for Idle Up 2.

`<ThrottleCurveA>` [i,i,i,i,i]

http://www.heli-x.net

Like `<ThrottleCurve>`, but for auto rotation.

`<NickExpo>` [i]
　　Expo settings for cyclic nick, -100 up to 100. Default is 0.

`<NickExpo2>` [i]
　　Like `<NickExpo>`, but for Idle Up 2.

`<NickExpoA>` [i]
　　Like `<NickExpo>`, but for auto rotation.

`<NickDualRate>` [i]
　　Dual rate settings for cyclic nick, 25 up to 125. Default is 100.

`<NickDualRate2>` [i]
　　Like `<NickDualRate>`, but for Idle Up 2.

`<NickDualRateA>` [i]
　　Like `<NickDualRate>`, but for auto rotation.

`<RollExpo>` [i]
　　Expo settings for cyclic nick, -100 up to 100. Default is 0.

`<RollExpo2>` [i]
　　Like `<RollExpo>`, but for Idle Up 2.

`<RollExpoA>` [i]
　　Like `<RollExpo>`, but for auto rotation.

`<RollDualRate>` [i]
　　Dual rate settings for cyclic roll, 25 up to 125. Default is 100.

`<RollDualRate2>` [i]
　　Like `<RollDualRate>`, but for Idle Up 2.

`<RollDualRateA>` [i]
　　Like `<RollDualRate>`, but for auto rotation.

Main rotor — thrust

`<MultiplicatorRotorLift>` [f]
    This factor is used to fit the thrust of the rotor.

`<MultiplicatorRotorLiftAutorotation>` [f]
    Same as `MultiplicatorRotorLift`, but valid during auto rotation.

`<MultiplicatorAirResistanceM>` [f]
    This factor determines the air resistance of the rotor which is independent of the angle of attack.

`<MultiplicatorAirResistanceN>` [f]
    This factor determines the air resistance of the rotor which is dependent on the angle of attack.

`<MultiplicatorStallLift>` [f]
    This factor controls the lift during stall.

`<FactorAirResistanceDuringStall>` [f]
    During stall you can influence the air resistance (dependent on pitch) with this factor.

`<StallAngle>` [f]
    This is the angle (in degree), for which a stall might occur.

`<StallAngleAutorotation>` [f]
    Like StallAngle, but valid during the autorotation phase.

`<StallRevPerMin>` [f]
    The stall can only happen when the number of revolutions is below this number.

`<MultiplicatorNickRollPower>` [f]
    This factor increases the power consumption caused by cyclic controls. It is due to the air resistance of the paddles.

`<AdditionalLinearAirResistance>` [f]
    With this parameter you can adjust the air resistance during forward flight (independent of the angle of attack).

`<AdditionalLinearPitchAirResistance>` [f]
    With this parameter you can adjust the air resistance during forward flight (dependent on the angle of attack).

http://www.heli-x.net

`<GroundEffectExponent>` [f]

Tuning of the ground effect. Default is 1. Higher values will increase the ground effect.

`<GroundEffectIntensity>`[f]

Tuning of the ground effect. Default is 1. Higher values will increase the horizontal instability.

`<TipLossFactor>` [f]

The thrust at the tips of the blades is not optimal. If you decrease the value (default is 1) you can decrease the thrust in order to simulate the tip losses.

`<VerticalClimbFactor>` [f]

This factor is very experimental. Values greater than the default value 1 will restrict the maximal climb speed of the helicopter. Maybe this parameter will be erased at some time.

`<MultiplicatorRotorTorque>` [f]

Factor to increase the torque caused by the main rotor.

`<EulerFlowMagicNumber1>` [f]

This parameters controls the inertia of the air flowing through the main rotor. This influences the reaction speed in pitch changes.

`<EulerFlowMagicNumber2>` [f]

This factor controls the flow situation in forward flight. It is responsible for the gain of lift during forward flight.

`<EulerFlowMagicNumber3>` [f]

This parameter controls the shape of the air flow above and below the main rotor. This controls e.g. the maximum climb velocity.

`<FactorVortexIntensity>` [f]

Strength of the vortex ring state.

`<FactorVortexDiffusion>` [f]

Diffusion constant of the vortex ring state.

`<FactorVortexInstability>` [f]

Instability of the vortex ring state.

Gyro — settings

`<YawRateMax>` [f]
　　Maximum rotation speed around the vertical axis in revolutions per second.

`<GyroGain>` [f]
　　Gyro gain.

`<GyroGain2>` [f]
　　Gyro gain for Idle Up 2.

`<GyroGainA>` [f]
　　Gyro gain during auto rotation.

`<GyroHeadingHoldGain>` [f]
　　Gain value for heading hold. 0 means no heading hold.

`<GyroHeadingHoldGain2>` [f]
　　Like `<GyroHeadingHoldGain>`, but for Idle Up 2.

`<GyroHeadingHoldGainA>` [f]
　　Like `<GyroHeadingHoldGain>`, but for Autorotation.

`<GyroSensorQuality>` [f]
　　For the simulation of the gyro drift. Value 0: no drift. Value greater than 0: Yes there is a drift.

`<YawExpo>` [i]
　　Expo setting for yaw, -100 up to 100. Default is 0.

`<YawExpo2>` [i]
　　Like `<YawExpo>`, but for Idle Up 2.

`<YawExpoA>` [i]
　　Like `<YawExpo>`, but for Autorotation.

`<YawDualRate>` [i]
　　Dual rate setting for yaw, 25 up to 125. Default is 100.

`<YawDualRate2>` [i]
　　Like `<YawDualRate>`, but for Idle Up 2.

http://www.heli-x.net

`<YawDualRateA>` [i]
>   Like `<YawDualRate>`, but for Autorotation.

3-Axes-Stabilisation system

`<StabiNickGain>` [f]
>   P-Factor for the control of cyclic nick.

`<StabiNickHeadingHoldGain>` [f]
>   I-factor for the control of cyclic nick.

`<StabiNickSensorQuality>` [f]
>   Quality of the sensor. 0 means best quality.

`<StabiNickRateMax>` [f]
>   The maximum rate in revolutions per second.

`<StabiRollNickForwardFactor>` [f]
>   The feed forward factor for nick, default value is zero, maximal value is 1.

`<StabiRollGain>` [f]
>   P-Factor for the control of cyclic roll.

`<StabiRollHeadingHoldGain>` [f]
>   I-factor for the control of cyclic roll.

`<StabiRollSensorQuality>` [f]
>   Quality of the sensor. 0 means best quality.

`<StabiRollRateMax>` [f]
>   The maximum rate in revolutions per second.

`<StabiRollFeedForwardFactor>` [f]
>   The feed forward factor for roll, default value is zero, maximal value is 1.

`<AccMaxAngle>` [f]
>   The maximum bank angle in ACC-Mode.

`<AccGainAngle>` [f]

Control factor (angle) in ACC-Mode.

`<AccGainOmega>` `[f]`
Control factor (angular velocity) in ACC-Mode.

Engine

`<EngineFrequencyRatio>` `[f]`
Frequency ratio of the engine to the main rotor.

`<OmegaTorqueData>` `[f,f,...]`
Table with the engine characteristics. Each pair represents a number of revolution per minute and the torque in Nm.

`<GearFrictionCoefficientNormal>` `[f]`
Friction coefficient for the friction which is proportional to the speed.

Fuselage

`<Face>` `[node]`
The fuselage is approximated using faces. Each face is represented by a list of the following tags:

`<Position>` `[f,f,f]`
The coordinates of the midpoint of the surface.

`<Normal>` `[f,f,f]`
The direction of the normal vector (pointing outwards) of the face.

`<CW>` `[f]`
The C_W-value if the face.

`<Area>` `[f]`
the area of the face in m^2.

Horizontal and vertical tails

`<VerticalTailArea>` [f]
    Area of the vertical tail plane.

`<VerticalTailNormal>` [f,f,f]
    A vector which is normal to the vertical tail plane. Often, 0,0,1 is correct.

`<VerticalTailPosition>` [f,f,f]
    The position of the midpoint of the vertical tail plane.

`<HorizontalTailArea>` [f]
    Area of the horizontal tail plane.

`<HorizontalTailNormal>` [f,f,f]
    A vector which is normal to the horizontal tail plane. Often, 0,1,0 is correct.

`<HorizontalTailPosition>` [f,f,f]
    The position of the midpoint of the horizontal tail plane.

Landing gear

`<ContactStiffness>` [f]
    Stiffness of the landing gear.

`<ContactViscosity>` [f]
    Damping of the landing gear in normal direction.

`<ContactViscosityTangential>` [f]
    Damping of the landing gear in tangential direction.

`<FrictionCoefficientX>` [f]
    Friction coefficient in x-direction, i.e. forwards.

`<FrictionCoefficientZ>` [f]
    Friction coefficient in z-direction, i.e. sidewards.

`<MaximalContactVelocity>` [f]
    The maximal velocity during contact so that there is no crash (in m/s). Default: 5.

Sound

`<SoundVolumeFactor>` [f]
Loudness factor of the engine sound, between 0 and 1.

`<BladeVolumeFactor>` [f]
Loudness factor of the blade sound, between 0 and 1.

`<CrashVolumeFactor>` [f]
Loudness factor of the crash sound, between 0 and 1.

`<SoundPitchFactor>` [f]
A factor which scales the influence of the pitch of the rotor. Default value is 1.

`<SoundMinPitch>` [f]
The influence of the pitch value is restricted. The minimum pitch is set by this factor. Default value is 0.5.

`<SoundPaxPitch>` [f]
The influence of the pitch value is restricted. The maximum pitch is set by this factor. Default value is 1.5.

`<BladePitchFactor>` [f]
Similar as `SoundPitchFactor` for the blade sound.

`<BladeMinPitch>` [f]
Similar as `SoundMinPitch` for the blade sound.

`<BladePaxPitch>` [f]
Similar as `SoundPaxPitch` for the blade sound.

`<SoundConeExponentMain>` [f]
Determines how sharp the directional cone for the blade sound is. 0 means: no directional dependence, 10 means very strnng directional dependence. Default value: 1.

`<SoundDirectionDependenceFactorMain>` [f]
Determines how much of the complete blade sound is directional dependent. 1 means: The whole sound is directional dependent, 0 means: no directional dependence. Default value: 0.7

`<SoundConeExponentTail>` [f]

> Determines how sharp the directional cone for the tail sound is. 0 means: no directional dependence, 10 means very strnng directional dependence. Default value: 1.

`<SoundDirectionDependenceFactorTail>` [f]

> Determines how much of the whole tail sound is directional dependent. 1 means: The complete sound is directional dependent, 0 means: no directional dependence. Default value: 0.7

Smoke

`<SmokeIntensityFactor>` [f]

> The intensity of the smoke. Usually values in the range 0.5 up to 5.

`<SmokeDurationSeconds>` [f]

> The life time of a smoke particle in seconds.

`<SmokeDiffusionConstant>` [f]

> The value defines how fast the smoke spreads out in the air.

`<SmokeTimeConstant>` [f]

> The time constant determined how fast the smoke follows the speed of the surrounding air. Values which are too small (less than 0.01) might cause instabilities in the smoke simulation.

`<SmokeColorRGB>` [f,f,f]

> The smoke color as RGB value.

`<SmokeSourcePosition>` [f,f,f]

> The position of the smoke source in model coordinates.

`<SmokeDirection>` [f,f,f]

> The initial direction of the smoke velocity.

`<SmokeEmissionVelocity>` [f]

> The speed of the smoke in m/s.

`<SmokeEmissionSpread>` [f]

This value determined the diameter of the cone of the smoke generation. Typical values are in the range 0.1 to 1.

`<SmokeBlendingPreset1to8>` `[i]`
Here you define one out of 8 presets how the blending is realized in terms of openGL.

`<SmokeTextureNumber>` `[i]`
You choose the texture of the smoke with an integer value between 0 and 5. Default value: 0.

`<Smoke2IntensityFactor>` `[f]`
The intensity of the smoke. Usually values in the range 0.5 up to 5. For the second smoke source.

`<Smoke2DurationSeconds>` `[f]`
The life time of a smoke particle in seconds. For the second smoke source.

`<Smoke2DiffusionConstant>` `[f]`
The value defines how fast the smoke spreads out in the air. For the second smoke source.

`<Smoke2TimeConstant>` `[f]`
The time constant determined how fast the smoke follows the speed of the surrounding air. Values which are too small (less than 0.01) might cause instabilities in the smoke simulation. For the second smoke source.

`<Smoke2ColorRGB>` `[f,f,f]`
The smoke color as RGB value. For the second smoke source.

`<Smoke2SourcePosition>` `[f,f,f]`
The position of the smoke source in model coordinates. For the second smoke source.

`<Smoke2Direction>` `[f,f,f]`
The initial direction of the smoke velocity. For the second smoke source.

`<Smoke2EmissionVelocity>` `[f]`
The speed of the smoke in m/s. For the second smoke source.

`<Smoke2EmissionSpread>` `[f]`

This value determined the diameter of the cone of the smoke generation. Typical values are in the range 0.1 to 1. For the second smoke source.

`<Smoke2BlendingPreset1to8>` [i]
Here you define one out of 8 presets how the blending is realized in terms of openGL. For the second smoke source.

`<Smoke2TextureNumber>` [i]
You choose the texture of the smoke with an integer value between 0 and 5. Default value: 0. For the second smoke source.

6.2 Airports

We distinguish two kinds of airports: photo realistic ones with a fixed position of the pilot and a 3D type where the pilot position is not fix.

6.2.1 Photo realistic airports

Airports are made from photo realistic panorama pictures. Here are the details:

- HELI-X uses the cubic format. Details can be found e.g. here: http://www.panoguide.com/howto/panoramas/types.jsp.

- Each airports consists out of 6 pictures. They have the format 2048×2048 (SkyBox), 1024×1024 (SkyBox1024) pixels or more (SkyBoxHD).

- The smaller format is in particular for poor graphics adapters.

- The whole scene is in a directory which has the name of the airport.

- The six pictures should be in a subdirectory. Allowed names are `1024`, `2048` and `4096`. An example: The airport Alvier is in the directory Alvier and there are six files with the following names: `Alvier_back.jpg`, `Alvier_front.jpg`, `Alvier_left.jpg`, `Alvier_right.jpg`, `Alvier_bottom.jpg`, `Alvier_top.jpg`.

- Furthermore, the file `Alvier.xml` is needed. In this file there are informations regarding the direction to the sun, the start position of the helicopters and the

height of the horizon.

In the xml file which belongs to the airport, the following tags can be found:

`<Name>` [s]
    The name of the airport.

`<Author>` [s]
    The name of the author.

`<WWW>` [s]
    A link to the web site of the author.

`<Comment>` [s]
    Comments about the airport.

`<HeightOfGround>` [f]
    The height of the ground in m. Typically 0.

`<HeightOfEyes>` [f]
    The height of the eyes above the ground (in meter). A usual value is 1.7. This
    height should be the height of the camera when the photos are taken.

`<StartPosition>` [f,f,f]
    The start position of the helicopter. The coordinate system is a right handed one
    with x and z on the bottom, and y pointing upwards. In order to find a suitable
    start position, you can use the option `-DheliX.statusformat=1` in the start
    script. In this case, the current position of the helicopter is shown in the status
    bar. Or you can use the matrix view.

`<LightDir>` [f,f,f]
    The direction of the sun rays is important for the creation of the shadows. You
    have to input the x-, y- and z-component of the vector which describes the direction
    of the sun rays. Since the sun shines from above, the y value should be negative.

`<ShadowLambda>` [f]
    Define the lambda value for the shadow. Typical values are between 0 and 1.

`<ShadowIntensity>` [f]
    Define the shadow intensity. Typical values are between 0 and 1.

http://www.heli-x.net

`<UseSunshine>` [s]
    `true` or `false`.Switch on and off the sun glare effect.

`<SunshineFactor>` [s]
    Strength of the sun glare effect. Default is 0.7.

`<SunFocusFactor>` [s]
    Width of the sun glare effect. Default is 1.

`<StartPosition0>`, `<StartPosition1>`, ... [f,f,f]
    With these tags you can define the start positions for all the helicopters in the
    multi player simulation.

`<AmbientSoundFile>` [s]
    The name of a sound file for the ambient sounds. It has to be in the ogg-Format.

`<UseStandardHeightOfGroundPlane>` True or false. You define, if the ground
    plane is seen as a crash object in this setting.

`<FrictionFactorGroundPlane>` [f]
    With this factor, you multiply the friction of the ground plane, default: 1.

`<TopographyFile>` [s]
    The file name of a topography file, where the crash objects are defined.

`<Track>` [s]
    The file name of a track file, see below.

A topography file is a file which defines crash objects. There are three kinds of crash
objects:

`<Triangle>` A triangle, defined by three points.

`<Parallelogram>` A parallelogram. It is defined like a triangle with three points.

`<Plane>` A flat plane of infinite size.

For each of these crash objects you can define

`<CrashSensitivityFactor>` This factor defines how dangerous it is to crash this
    object, default: 1.

http://www.heli-x.net

<FrictionFactor> A factor, to increase ($>$ 1) or decrease ($<$ 1) the friction.

For triangles and parallelograms, there are in addition:

<Alpha> The Alpha value defines the opaqueness of the part. If Alpha is 1, the helicopter is not visible when behind the object. If Alpha is 0, the crash object is not visible. You can also use skybox pictures with alpha channel (then you have to use png files).

<Thickness> The thickness of the triangle. If the thickness is greater than zero, the helicopter can also collide from behind.

<CeilingType> Is the surface of type CeilingType, then the surface is ignored in the calculation of the start height.

<GroundType> A Surface of type GroundType is so that no object can be behind this object.

Example

- A triangle which is part of the bottom of the scene:
  GroundType=yes, CeilingType=no, Alpha=1, ShadowReceiver=yes

- A wall where you can also fly behind:
  GroundType=no, CeilingType=no, Alpha=1, ShadowReceiver=yes

- A fence which is partly transparent (made with an alpha channel in the texture):
  GroundType=no, CeilingType=no, Alpha=0.5, ShadowReceiver=yes oder no

- A Wall which is behind another wall, so that it can not be seen. But it is important for the collision detection:
  GroundType=no, CeilingType=no, Alpha=0, ShadowReceiver=no

- The roof of a house which is so high that you cannot see it from above:
  GroundType=yes, CeilingType=no, Alpha=0, ShadowReceiver=no

- The cap of a drain which is higher than the surrounding:
  GroundType=no, CeilingType=no, Alpha=1, ShadowReceiver=yes

- A Wall which is far away:
  GroundType=no, CeilingType=no, Alpha=1, ShadowReceiver=no

6.2.2 Scene3D–Airports

Airports of type `Scene3D` allow a free pilot position (for this, FPV has to be activated in the camera settings). In addition, the following tags are allowed.

`<VariablePilotPosition>` Set to true, if the pilot is allowed to move.

`<MaxPilotRadiusGround>` The maximum radius the pilot can move as long as he is on the ground.

`<MaxPilotRadiusAir>` The maximum radius the pilot can move when he is in the air.

`<Object>` An object is a graphical model in the wavefront format. Defining an object, you can define the following attributes:

    `<MakeCrashObject>` This defines, if the object is included in the collision and contact detection.

    `<CeilingType>` If a model is of ceiling type it is not considered when defining the height of the start position and in the ground effect.

    `<GroundType>` When a model is of ground type then internally `setDepthBufferWriteEnable` is set to false.

    `<FrictionFactor>` The friction factor gives the friction a factor of the usual friction coefficient.

Within an object description, the following tags have to be defined:

    `<Path>` The path to the wavefront file.

    `<Translation>` A translation in x, y, and z direction

    `<ShadowOnGround>` This determines whether a shadow is calculated for this object.

http://www.heli-x.net

<Rotation> The rotation angle (in degrees) around the *y* axis. In the rotation tag, you might set the attribute `Billboard`.

<AutoTextureShapeName> This defines the name of a group where the texture is variable.

<AutoTexturePath> With several of these tags the different paths of the textures are defined.

<AutoTextureSeconds> The time in seconds in which the textures are changed.

<ObjectFile> With this tag, you define another xml file where you can put object tags.

6.3 Scenes

A scene describes an extension of an airport. Thus, all objects, etc, known from the airports, can be used here. In addition, we have the following tags:

<EnvironmentToLoad> formats,s The environment which is the base for this scene.

6.4 Tracks

A track consists of a sequence of track objects. The following types are available:

Circle, `Type=Circle`

<IndexList> [n,...]
    The indices specify the order in which the objects have to be passed. Start index: 1.

<Translation> [f,f,f]
    The position of the object. If `Relative="true"` then the height is relative to the topography.

`<Direction>` [f,f,f]

    The vector describes in which direction the circle has to be passed.

`<Radius>` [f]

    The radius of the circle.

`<MarkerHeight>` [f]

    The height of the marker. If `Relative="true"` then the height is relative to the topography.

Pylon, `Type=Pylon`

`<IndexList>` [n,...]

    The indices specify the order in which the objects have to be passed. Start index: 1.

`<Translation>` [f,f,f]

    The position of the object. If `Relative="true"` then the height is relative to the topography.

`<RayDirection>` [f,f,f]

    The vector describes a ray which has to be passed.

`<Height>` [f]

    The maximal height which is allowed to fly around thy pylon.

`<ClockWise>` [b]

    The direction to fly around the pylon.

`<MarkerHeight>` [f]

    The height of the marker. If `Relative="true"` then the height is relative to the topography.

Gate, `Type=Gate`

`<IndexList>` [n,...]

    The indices specify the order in which the objects have to be passed. Start index: 1.

http://www.heli-x.net

`<Translation> [f,f,f]`
　　The position of the object. If `Relative="true"` then the height is relative to the topography.

`<Direction> [f,f,f]`
　　The vector describes in which direction the circle has to be passed.

`<HorizontalHalfAxis> [f]`
　　The half width of the gate.

`<VerticallHalfAxis> [f]`
　　The half height of the gate.

`<MarkerHeight> [f]`
　　The height of the marker. If `Relative="true"` then the height is relative to the topography.

Hover Area, `Type=HoverArea`

`<IndexList> [n,...]`
　　The indices specify the order in which the objects have to be passed. Start index: 1.

`<Translation> [f,f,f]`
　　The position of the object. If `Relative="true"` then the height is relative to the topography.

`<Radius> [f]`
　　The radius of the hover area.

`<MinimumHeight> [f]`
　　The minimal height during hovering.

`<MaximumHeight> [f]`
　　The maximal height during hovering.

`<Duration> [f]`
　　The number of seconds for the hovering. When the color gets yellow, the time runs.

`<MarkerHeight>` [f]
> The height of the marker. If `Relative="true"` then the height is relative to the topography.

Directional Hover Area, `Type=DirectionalHoverArea`

`<IndexList>` [n,...]
> The indices specify the order in which the objects have to be passed. Start index: 1.

`<Translation>` [f,f,f]
> The position of the object. If `Relative="true"` then the height is relative to the topography.

`<Direction>` [f,f,f]
> The direction in which the hovering has to be done.

`<DirectionAccuracyDegrees>` [f]
> The required accuracy for the direction during hover.

`<Radius>` [f]
> The radius of the hover area.

`<MinimumHeight>` [f]
> The minimal height during hovering.

`<MaximumHeight>` [f]
> The maximal height during hovering.

`<Duration>` [f]
> The number of seconds for the hovering. When the color gets yellow, the time runs.

`<MarkerHeight>` [f]
> The height of the marker. If `Relative="true"` then the height is relative to the topography.

Rotating Hover Area, `Type=RotatingHoverArea`

`<IndexList> [n,...]`
    The indices specify the order in which the objects have to be passed. Start index: 1.

`<Translation> [f,f,f]`
    The position of the object. If `Relative="true"` then the height is relative to the topography.

`<Direction> [f,f,f]`
    The direction in which the hovering has to be done.

`<DirectionAccuracyDegrees> [f]`
    The required accuracy for the direction during hover.

`<Radius> [f]`
    The radius of the hover area.

`<MinimumHeight> [f]`
    The minimal height during hovering.

`<MaximumHeight> [f]`
    The maximal height during hovering.

`<Duration> [f]`
    The number of seconds for the hovering for each step. When the color gets yellow, the time runs.

`<Steps> [n]`
    The number of steps for the rotating process.

`<ClockWise> [b]`
    The direction of the rotation.

`<MarkerHeight> [f]`
    The height of the marker. If `Relative="true"` then the height is relative to the topography.

Landing Area, `Type=LandingArea`

`<IndexList> [n,...]`
　　The indices specify the order in which the objects have to be passed. Start index: 1.

`<Translation> [f,f,f]`
　　The position of the object. If `Relative="true"` then the height is relative to the topography.

`<Radius> [f]`
　　The radius of the landing area.

`<MarkerHeight> [f]`
　　The height of the marker. If `Relative="true"` then the height is relative to the topography.

6.5 Utilities

It is possible, to load utilities in an airport. Airport and utilities are then called a scene. Currently, there are three types of utilities: Boxes, pylons and heliports. They differ in their creation of shadows and in their collision behavior. You can put the utilities at any position in the scene. As always: The coordinates of the ground are x and z.

The xml file for the utilities consists out of the following tags:

`<PhysicalModelName> [s]`
　　The name of the internal simulation model. You have the choice: `Pylon`, `Heliport` or `Box`.

`<Author> [s]`
　　The name of the author.

`<WWW> [s]`
　　The URL of the author's web site.

`<Comment> [s]`
　　Comments about the model

`<Scale> [f]`
　　This factor scales from the model coordinates to the world coordinates. The

default value is 1.

7 Release Notes

Version 10.1   New in Version 10.1 compared to 10.0:

- Improved helicopter phyiscs

- More realistic wind physics

- New sun glare

- Spinning wheels simulation

- Vanishing wheels for the Comet

- Zoomed model support in videos

- Improved color changes in models

- New text HUDS (Bank angle, pitch angle)

- Moving stick display

- Multi pilot position online airports

- Spectator mode

- Faster listing of resources for download

- Improved matrix view for setting the sun position

- Minor bug fixes

Version 10.0   New in Version 10.0 compared to 9.2:

- Based on a newer version of the graphics library

- Support for virtual reality (VR)

http://www.heli-x.net

- Support for Apple's M1 processor

- Airports of type 4K with a higher resolution

- Better graphics by physical based rendering

- Water shader included

- The controllers can now be changed while HELI-X is running

- Improved configuration for switches and keyboard functions.

- Use of the actual position as place for the trainings

Version 9.2   New in Version 9.2 compared to 9.1:

- Auggie's instinctive training ring

- Function for the reset of the model

Version 9.1   New in Version 9.1 compared to 9.0:

- New HUDs for Position and height, possibility for zoom

- New text HUD for bank

- Speech sound for chosen bank (with voices from VBar)

- Butterfly ailerons

- V-shape for elevator

- Separate controller configurations for planes or helicopters

- Improved ambient light for airports

- Use of VBar touch with WLAN

- Favorite start positions

- Three banks instead of two

- Object zoom for own model

- Improvements for sound

Version 9.0   New in Version 9.0 compared to 8.0:

Global

- 4k resolution by larger fonts

- Several start positions to choose from

- Hand launch also with 50, 100, and 150m

- Pilot positions selectable in 3d scenes

- Favorites also definable in the selection windows

- Slow Motion switchable with the keyboard

- Ghost sticks

- New simple stick HUD

- 4in1 stick HUD

- Definitions of video segments

- Save video segments as new video

- Vertical adaptive view in camera

- HUDs in text form

- Calibration button for controller configuration

http://www.heli-x.net

- Split screen vertical and horizontal

Models

- Improved physical models for helicopters, planes, and multicopters

- CoG editable

- Folding-out turbines (implemented in the Swift)

- New electric and nitro power trains

- New sound systems

- Switchable smoke generators (implemented in the Swift)

- Reworked downwash

- Variable pitch props

- Ground effect improved

- Improved Stabilisation system

- Rotor disc visualisation improved

- Improved downwind of wing

- Only wing aerodynamics improved

- Pitch dynamics and feed forward in flight phases

Training

- Replay training with possibility to save the snippets as HELI-X recording

- Music training

Multiplayer

- Server with port mapping possible

http://www.heli-x.net

Version 8.0 New in Version 8.0 compared to 7.0:

- Plane models are available.

- More flight modes for drones.

- Controller setting now supports push buttons.

- Trim settings.

- Torque Training

- Emergency landing training

- Spin Training

- Supoort of the StickMover

- New HUD display (Simple HUD)

- HUD for gear and flaps status.

- HUD for stall warning.

- Possibility for hang soaring

- And many other changes to improve HELI-X

Version 7.0 New in Version 7.0 compared to 6.1:

- Improved sound of the helicopters

- Improved graphics of the rotos

- Improved scenes concept

- Tracks for Training and FPV race

- Dual cameras and split views

http://www.heli-x.net

- Stick display improved

- Timer

- Video–Export with highest quality

- And many other changes to improve HELI-X

Version 6.1   New in Version 6.1 compared to 6.0:

- New physics for collective pitch coaxial helicopters

- Rescue mechanism

- New FPV camera mode (leveled pilot's view)

- New head up display (HUD)

- Adjustable rotor transparency

- Hover training with swing mode

- And many other changes to improve HELI-X

Version 6.0   New in Version 6.0 compared to 5.0:

- Completely openGL-based, no use of Swing anymore

- Possibility of full screen display

- Sun glare and anisotropic texture filter

- Improved aerodynamics

- More accurate simulation of the ground effect

- Improved engine rotor coupling

http://www.heli-x.net

- Crash and retry Training

- Detail window

- Camera with keep–ground–in–view

- Improved and new head up displays

- Gear like animations at helicopters

- And many other changes to improve HELI-X

Version 5.0   New in Version 5.0 compared to 4.1:

- Use of a better graphics engine, based on OpenGL 2.0. Shader support

- Better shadows, better graphics.

- Improved airport selection mechanism

- 3D CP-quadrocopters

- Animations of helicopter parts, retractable gear, etc.

- And many other changes to improve HELI-X

Version 4.1   New in Version 4.1 compared to 4.0:

- Multicopter (Tricopter, Quadrocopter, Hexacopter)

- Acro- and ACC-Mode for Multicopter

- Improved ground effect

- Improved collision handling

- Flight path for a better training

http://www.heli-x.net

- Favorites and ignore function

- Japanese and Russian language

- Some minor changes and bug fixes

Version 4.0   New in Version 4.0 compared to 3.0:

- Download manager for the installation of models, airports, scenes, videos

- improved physics

- physics for coaxial helicopters

- controller curves with dead zone in the middle for weaker controllers

- reaction training

- no collective training

- promenade training

- funnel training

- wind display

- better rotor reflections, dependent on the pitch value

- elastic blades visualization of the rotating rotor

- new models

- new 3D scene Heli Flying field

- better performance in multiplayer simulation

- rework of the keyboard and button functions

- Polish and Italian Language support

http://www.heli-x.net

- some bug fixes

Version 3.0   New in Version 3.0 compared to 2.0:

- Crash-Objekte in foto scenes

- 3D scenes

- Pilot position vaiable (in 3d scenes)

- FPV-View (in 3D scenes)

- Smoke

- Tail rotor sound

- ambient sound

- multiplayer using color and texture information

- Basket training

- Rope training

- Failure training

- variable vertical helicopter position on screen

- Object zoom

- Performance updates

- History for helicopter and airport selection

- Filter for the helicopter selection

- Plugins

- Statistics information

http://www.heli-x.net

- Online news

- New helicopter models

- Czech and Norwegian language supported

Version 2.0   New in Version 2.0 in contrast to 1.1:

- Update of the helicopter physics

- Simulation of three axes gyro systems

- Import of V-Stabi Settings

- New dynamic objects (boxes, balls, etc.)

- Balloon training

- Cargo training

- Cage training

- Display of a position marker

- Throttle and pitch curves can be stored

- Slow motion

- French version

- Possibility of a game server (already in the demo version)

- Better adjustment to the performance of the hardware

- New helicopters, textures with alpha channel

- Helicopter colors can be changed interactively

http://www.heli-x.net

Version 1.1   New in Version 1.1 in contrast to 1.0:

- Pitch- and throttle-curves, Possibility for fixed pitch helis

- Gyro settings dependent on flight state

- Improvements of the gyro algorithms. Simulation of the tail servo

- Auto rotation training

- Tail training

- Stall-meter

- Slip-meter

- Rotor head instability

- Improvement of the hover training

- New keyboard functions

- Automatic screen saver deactivation

- Improvements of the graphics of rotating parts

- Change of the view during multi player not only via the menu but also by click in the pilot window

- New helis: T-Rex 600, Honey Bee King V2, Dragonus