

Установка «продвинутого» ПО для Kuka Youbot

Антонов Е. С.

antonovspostbox@gmail.com

4 января 2017 г.

Введение

Не все версии пакетов `youbot_driver` и `youbot_driver_ros_interface` могут работать с манипулятором, если к нему не подключен его стандартный gripper¹. Причем речь идет и о той ситуации, когда последний активирован в конфигурационном файле², и о той, когда он с его помощью отключен так, как это описано в разделе «Modify Software» [здесь](#). В первом из этих случаев нода `youbot_driver_ros_interface` из одноименного пакета при ее запуске завершается аварийно, во втором — не завершается и даже манипулятор калибрует, однако потом отказывается его подключить, мотивируя это тем, что `gripper is disabled`.

Происходит, как я понял, сказанное из-за того, что грамотная обработка ситуации отсутствия gripper-а на работе попросту не реализована нормально/{в полной мере}. В качестве иллюстрации к своим словам могу привести следующий факт. При своей работе нода `youbot_driver_ros_interface` опирается на значение аргумента `enableStandardGripper` функции `initializeArm()` (см. например, [эту](#) и [эту](#) строчки). Последнее же как задается [при инициализации этой функции](#) по умолчанию равным `true`, так и не меняется потом нигде (см. Приложение [А](#)).

Я нашел два способа устранить эту проблему и, следовательно, запустить манипулятор без gripper-а так, чтобы он потом нормально или почти нормально работал. Их описание приведено ниже по тексту.

¹Например, на это не способны те их версии, которые распространяются из стандартного репозитория и становятся доступными после установки командами

```
$ sudo apt-get install ros-indigo-youbot-driver
```

```
$ sudo apt-get install ros-indigo-youbot-driver-ros-interface
```

²По умолчанию это `youbot-manipulator.cfg` из папки `config` пакета `youbot_driver`.

Первый способ («грязный»)

Первый вариант решения обозначенной проблемы заключается в том, чтобы поменять значение уже упомянутого параметра `enableStandardGripper` [здесь](#) с `true` на `false` и после этого пересобрать пакет `youbot_driver_ros_interface`. После этого в случае, если `gripper` указанным [здесь](#) способом был отключен в конфигурационном файле, манипулятор нормально запустится и будет исправно работать, однако терминал будут заваливать предупреждающие сообщения (`warnings`) о том, что с `gripper`-ом что-то не так. Если попытаться запустить манипулятор, не отключив его в конфигурационном файле, то данный метод не поможет (манипулятор не запустится).

Подводя итог данному методу, можно сказать, что его несомненным плюсом является простота, а минусом — то, что обнаружен он был методом «тыка», и о гарантии стабильности его работы поэтому речи быть не может.

Второй способ

Второй способ решения рассматриваемой проблемы состоит в использовании «самых свежих» наработок в области данного ПО. Имеется в виду следующее.

Если обратиться на [Github-овскую страницу пакета youbot_driver_ros_interface](#), то в разделе «Pull requests» можно увидеть, что один из них имеет наименование «Add gripper availability check». Просмотрев предлагаемые входящими в его состав `commit`-ами изменения, можно обнаружить, что он предоставляет решение как раз таки нашей проблемы — обеспечение нормального запуска манипулятора в независимости от того, подключен `gripper` или нет.

С одной стороны, это победа, с другой, если попробовать применить эти изменения к своей копии пакета `youbot_driver_ros_interface`, скорее всего ничего не получится. Процесс компиляции завершится с ошибкой:

```
'class youbot::YoubotManipulator' has no member named 'hasGripper'
```

Сказанное произойдет по той причине, что изменения, вносимые выше упомянутым `pull request`-ом, требуют наличия в исходном коде пакета `youbot_driver` некоторых вещей (например, функции `hasGripper`), которые есть только в ветке `hydro-devel` относящегося к последнему [репозитория](#) и, таким образом, отсутствуют в ветке `master` и той версии пакета `youbot_driver`, которая добывается установкой «через `apt-get`».

Если же установить ту версию пакета `youbot_driver`, которая представлена в ветке `hydro-devel`, то все становится на свои места: пакеты `youbot_driver` и `youbot_driver_ros_interface` начинают собираться без ошибок, а манипулятор под их управлением — работать всегда, другими словами в независимости от того подключен или отключен gripper физически и/или программно.

Подводя итог описанию данного метода, хочется еще раз, но уже пошагово, обозначить упоминаемые в нем действия. Итак, согласно данному методу необходимо:

1. удалить установленные на компьютере версии пакетов `youbot_driver` и `youbot_driver_ros_interface`;
2. получить исходники ветки `hydro-devel`³ пакета `youbot_driver` одним из двух способов:
 - 1-ый способ:
 - (a) пройти по [этой ссылке](#);
 - (b) нажать на зеленую кнопку с фразой «Clone or download»;
 - (c) нажать в появившемся окне на кнопку «Download ZIP»;
 - (d) переместить из скачанного архива лежащую в нем папку `youbot_driver-hydro-devel` в папку `src` используемого ROS-овского workspace;
 - (e) удалить из имени папки (пакета) окончание «-hydro-devel».
 - 2-ой способ:
 - (a) зайти в терминале в папку `src` используемого ROS-овского workspace:

```
$ cd {path_to_your_workspace}/src
```
 - (b) клонировать с Github репозиторий `youbot_driver`:

```
$ git clone https://github.com/youbot/youbot_driver.git
```
 - (c) перейти в папку `youbot_driver`:

```
$ cd youbot_driver
```

³Стоит отметить, во-первых, что эта версия пакета собирается и на ROS Indigo, а во-вторых, что его ветка `hydro-devel` является чисто опережающей по отношению к `master` (содержит все, что содержится в `master`, при этом `master` не содержит все, что содержится в ней). Увидеть последнее можно, например, [здесь](#).

- (d) создать в склонированном репозитории локальную ветку `hydro-devel`, отслеживающую одноименную ей из Github-овского репозитория, и перейти к ней:

```
$ git checkout -b hydro-devel origin/hydro-devel
```

3. собрать пакет `youbot_driver`;

4. получить исходники пакета `youbot_driver_ros_interface`, учитывающие изменения выше указанного [pull request-a](#):

- (a) зайти в папку `src` используемого ROS-овского workspace:

```
$ cd {path_to_your_workspace}/src
```

- (b) клонировать с Github репозиторий `youbot_driver_ros_interface`:

```
$ git clone https://github.com/youbot/youbot_driver_ros_interface.git
```

- (c) перейти в папку `youbot_driver_ros_interface`:

```
$ cd youbot_driver_ros_interface
```

- (d) создать на основе коммитов [pull request-a](#) содержащую их ветку (в примере она будет именоваться `cool-request`)⁴:

```
$ git fetch origin pull/13/head:cool-request
```

- (e) переключиться на созданную ветку:

```
$ git checkout cool-request
```

5. собрать пакет `youbot_driver_ros_interface`.

После установки данных пакетов рекомендуется обратиться к Приложению [В](#).

⁴Команда взята [отсюда](#).

Приложение А

Текстовый поиск в файлах с исходным кодом

Текстовый поиск в файлах с исходным кодом может быть полезен тогда, когда необходимо найти все места в последнем, где так или иначе используется некоторая переменная, функция и т.д. .

В git-е он реализован в виде команды⁵

```
$ git grep {search string}
```

осуществляющей поиск строки "search string" в файлах с исходным кодом текущего репозитория.

К слову сказать, у меня на компьютере данная комбинация команд⁶

```
$ roscd youbot_driver_ros_interface
```

```
$ git checkout indigo-devel
```

```
$ git grep -n enableStandardGripper
```

дает следующий вывод-результат:

```
include/youbot_driver_ros_interface/YouBotOODLWrapper.h:109: * @param
    enableStandardGripper If set to true, then the default gripper of the youBot
    will be initialized .
include/youbot_driver_ros_interface/YouBotOODLWrapper.h:111: void
    initializeArm (std :: string armName, bool enableStandardGripper = true);
src/YouBotOODLWrapper.cpp:118:void YouBotOODLWrapper::initializeArm(std::
    string armName, bool enableStandardGripper)
src/YouBotOODLWrapper.cpp:153:     if (enableStandardGripper)
src/YouBotOODLWrapper.cpp:204: if (enableStandardGripper)
```

⁵Подробнее о текстовом поиске в git читайте [тут](#).

⁶Ключ `-n` добавляет к выводу номера строк, в которых находится "search string".

Приложение В

Важные мелочи

При установке пакетов `youbot_driver` и `youbot_driver_ros_interface` легко можно забыть о двух важных вещах. Во-первых, о том, что после их сборки в отношении ноды `youbot_driver_ros_interface` последнего необходимо выполнить следующую команду⁷:

```
$ sudo setcap cap_net_raw+ep "{path_to_your_workspace}/devel/lib/\
youbot_driver_ros_interface/youbot_driver_ros_interface"
```

во-вторых, о том, что при запуске одного только манипулятора начало конфигурационного файла `youbot-manipulator.cfg` из пакета `youbot_driver` должно иметь следующий вид⁸:

```
[JointTopology]
ManipulatorJoint1 = 1
ManipulatorJoint2 = 2
ManipulatorJoint3 = 3
ManipulatorJoint4 = 4
ManipulatorJoint5 = 5
```

⁷Источники: [1](#) и [2](#).

⁸См. раздел «Using arm only» [здесь](#).