Schedulers for OCP applications

1. Description

The particular functionality aims to isolate the task scheduling functionalities from applications that need to perform scheduling decisions. Instead, the scheduler is a dedicated server (to be thought of as a daemon running at some peer) which can assist applications by monitoring the execution of tasks and recommending them scheduling decisions.

2. Components

The main component that provides the particular functionality is the scheduler. Schedulers run as standalone daemons at some peers and are available to OCP applications which, in order to optimize the access to resources, need to perform some efficient task scheduling.

The scheduler runs a particular protocol which consists of the following four states:

- The IDLE state: a scheduler at this state is available to be used by any application
- The ACTIVE state: a scheduler is in this state if it has already been assigned to some application
- The ONLINE state: a scheduler is in this state if it has already started serving clients (tasks) in an online manner
- The OFFLINE state: a scheduler is in this state if it has already started collecting tasks in order to compute an offline schedule for them

All other components that participate in this functionality are clients which issue commands to the scheduler. There are three different kinds of clients:

- Application clients: these are applications run on top of the OCP that use the scheduler in order to perform some efficient task scheduling. Commands that are sent to the scheduler by an application client include requesting/releasing the scheduler, starting online or offline scheduling, stopping an online scheduling process or scheduling collected tasks in an offline manner, and requesting scheduling reports.
- Machine clients: these are applications run on top of the OCP which control the access to resources of the same kind. These will be the resources on which the scheduler controls during the scheduling process. They can send commands to the scheduler declaring their will to participate in or abort the scheduling process as resources or terminate their participation, and report events such as arrival of a new task, termination of a task, status information, etc.
- Task clients: these are applications run on top of the OCP which request service from the scheduler. They can send commands requesting service, commands requesting information on the available resources, and commands requesting migration from a current machine client due to unsatisfactory service.

3. The protocol

Initially, the scheduler is in IDLE state. The protocol has the following simple structure which consists of simple communication rounds. In each communication, the scheduler waits for messages by a client, processes them by performing the corresponding actions, and replies with another message. The messages that can be sent by the scheduler are either OK or error messages (ERR).

The messages that can be received by the scheduler are presented in the following. A graphical representation of the protocol is depicted in Fig. 1.

3.1 Request messages (REQ)

This message is send by application clients which use it in order to use the scheduler. The message has the following format: `REQ#`

Unless the scheduler is in an IDLE state, the scheduler replies with an error message:

`ERR#REQ message received on state <state>`

When in IDLE state, the scheduler replies with an OK message and sets its state to ACTIVE.

3.2 Release messages (REL)

This message is also issued by application clients which use it in order to release a previously allocated scheduler. The message has the following format: `REL#`

Unless the scheduler is in ACTIVE state, the scheduler replies with an error message:

`ERR#REL message received on state <state>`

When in ACTIVE state, the scheduler replies with an OK message and sets its state to IDLE.

3.3 New-machine messages (MAC)

This message is issued by machine clients in order to declare that they aim to participate as machines in scheduling. The message has the following format:

`MAC#<machine name> <machine speed>`

When the scheduler is in IDLE state, it replies with an error message:

`ERR#MAC message received on state IDLE`

Otherwise, the scheduler registers the corresponding machine and replies with an OK message.[1]

3.4 Start-online-scheduling messages (SON)

This message is also issued by application clients which use it in order to command the scheduler to start scheduling tasks in an online manner. The message has the following format:

`SON#<algorithm> <monitoring> <parameters>`

The argument `<algorithm>` indicates the scheduling algorithm to run and the argument `<parameters>` contains parameters used by the algorithm. The `<monitoring>` option can be ON or OFF and denotes whether the scheduler monitors the actual load of machines or not.

Unless the scheduler is in ACTIVE state, it replies with an error message:

`ERR#SON message received on state <state>`

When in active state, if no machines have registered then the scheduler will issue an error message:

`ERR#no machine has registered`

If there are registered machines, the scheduler sets its state to ONLINE and replies with an OK message:

[1] Error messages can be returned if machine registration fails. To be completed.

```
OK #ONLINE state
```

3.5 Start-offline-scheduling messages (SOF)

This message is also issued by application clients which use it in order to command the scheduler to start collecting tasks in order to schedule them later in an offline manner. The message has the following format:

```
SOF#<algorithm>
```

Unless the scheduler is in ACTIVE state, it replies with an error message (ERR). When in active state, it either issues an error message (ERR) if no machines have registered, or replies with an OK message and sets its state to OFFLINE.

3.6 New-task messages (JOB)

The message is issued by task clients in order to declare that they aim to run a task on some machine. The message has the following format:

```
JOB#<task name> <task load> <#machines> <mach_1> <mach_2> .. <mach_m>
```

Unless the scheduler is in ONLINE or OFFLINE state, the scheduler replies with an error message (ERR). When in OFFLINE state, the scheduler registers the corresponding task and replies with an OK message. When in ONLINE state, the scheduler computes a scheduling for the particular task and replies with an OK message.

3.7 Information request messages (SHM)

This message is issued by task clients in order to command the scheduler to send them information on the available machines. The message has the following format:

```
SHM#
```

Unless the scheduler is in ONLINE or OFFLINE state, the scheduler replies with an error message (ERR). When in ONLINE or OFFLINE state, the scheduler replies with an OK message that contains the list of the available machines and their characteristics.

3.8 Stop-scheduling messages (END)

This message is also issued by application clients which use it in order to command the scheduler to stop scheduling tasks. The message has the following format:

```
END#
```

Unless the scheduler is in ONLINE or OFFLINE states, it replies with an error message (ERR). When in ONLINE or OFFLINE state, it replies with an OK message and sets its state to ACTIVE.

3.9 Schedule-all messages (ALL)

This message is also issued by application clients which use it in order to command the scheduler to schedule the collected tasks in an offline manner. The message has the following format:

```
ALL#
```

Unless the scheduler is in OFFLINE state, it replies with an error message (ERR). When in OFFLINE state, it either replies with an error message (ERR) if no tasks have been collected, or computes the schedule for the collected tasks, and replies to the application client with an OK message.

3.10 Migration-request messages (MIG)

This message is issued by task clients in order to notify the scheduler about unsatisfactory service and request rescheduling. The message has the following format:

```
MIG#<task name> <machine>
```

Unless the scheduler is in ONLINE state, the scheduler replies with an error message (ERR). When in ONLINE state, the scheduler replies with an OK message that contains the new scheduling recommendation.

3.11 Task-cancel messages (JCL)

This message is issued by task clients in order to notify the scheduler that they do not wish to be scheduled anymore. The message has the following format:

```
JCL#<task name> <machine>
```

Unless the scheduler is in ONLINE or OFFLINE state, the scheduler replies with an error message (ERR). When in ONLINE or OFFLINE state, the scheduler replies with an OK message.

3.12 Machine-abort messages (MAB)

This message is issued by machine clients in order to notify that they terminate their participation as machines in scheduling. The message has the following format:

```
MAB#<machine name>
```

When the scheduler is in IDLE state, it replies with an error message:

```
ERR#MAB message received on state IDLE
```

Otherwise, the scheduler unregisters the corresponding machine and replies with an OK message.

3.13 Task-start messages (JST)

This message is issued by machine clients in order to notify the scheduler that a new task begins. The message has the following format:

```
JEN#<machine name> <task name>
```

Unless the scheduler is in ONLINE state, it replies with an error message:

```
ERR#JST message received on state <state>
```

Otherwise, if the monitoring option is ON, the scheduler updates its information about the current schedule and replies with an OK message. In the case that the monitoring option is OFF, the scheduler replies with an error message (ERR) ignoring the information in the JST message.

3.14 Task-end messages (JEN)

This message is issued by machine clients in order to notify the scheduler that the execution of a task finished. The message has the following format:

```
JEN#<machine name> <task name>
```

Unless the scheduler is in ONLINE state, it replies with an error message:

```
ERR#JEN message received on state <state>
```

Otherwise, if the monitoring option is ON, the scheduler removes the particular task from the current schedule and replies with an OK message. In the case that the monitoring option is

OFF, the scheduler replies with an error message (ERR) ignoring the information in the JEN message.

3.15 Query messages (QRY)

This message is issued by task clients in order to query the scheduler for the latest scheduling decision concerning a particular task. The message has the following format:

```
QRY#<task name>
```

Unless the scheduler is in OFFLINE state, the scheduler replies with an error message (ERR). When in OFFLINE state, the scheduler replies with an OK message that contains information about the scheduling decision for the particular task.

3.16 Report messages (REP)

This message is issued by application clients in order to notify the scheduler about their current load. The message has the following format:

```
REP#
```

The scheduler replies with an OK message containing information about the current schedule.

```
OK#...
```

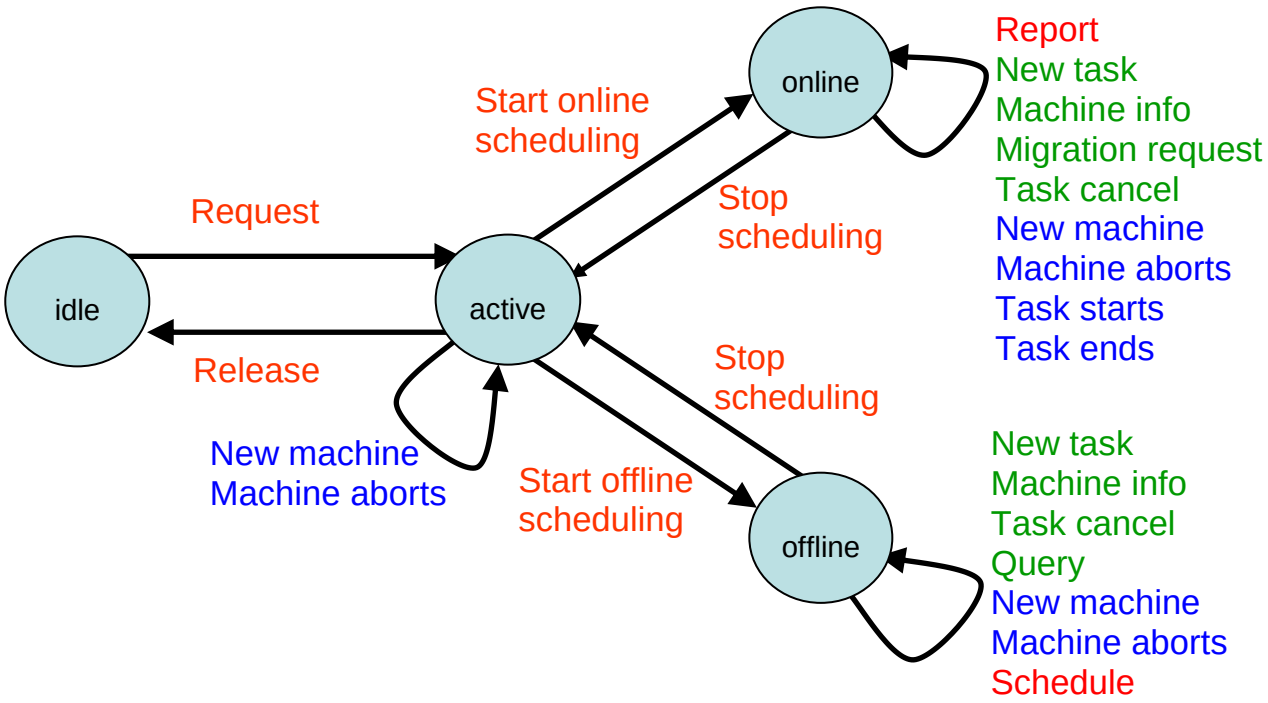

Figure 1. The protocol of the scheduler. Red labels denote messages issued by application clients, blue labels denote messages issued by machine clients, and green labels denote messaged issued by task clients. Arrows represent the change of state on success.

4. Scheduling algorithms

The current version supports the following online algorithms for minimizing the Lp norm of the machine loads and the makespan:

- FASTEST: each task is assigned to the fastest permissible machine.
- GREEDYINFNORM: each task is assigned to that permissible machine so that the maximum load over all machines is minimized.
- GREEDYLPNORM: each task is assigned to that permissible machine so that the Lp norm of the machine loads is minimized. The parameter p is an integer passed through the argument `<parameter>`.
- MAKESPAN: each task is assigned to that permissible machine so that the Lp norm of the machine loads is minimized, with p equal to the natural logarithm of the number of available machines.