
VMO-Score Documentation

Release 0.1.0

**Jaime Arias,
Myriam Desainte-Catherine,
and Shlomo Dubnov**

May 20, 2017

Contents

1	VMO_Score package	3
1.1	VMO_Score.segmentation	3
1.2	VMO_Score.petri_net	4
1.3	VMO_Score.improviser	6
1.4	VMO_Score.utils	7
2	Indices and tables	9
	Python Module Index	11

Contents:

CHAPTER 1

VMO_Score package

<i>VMO_Score.segmentation</i> (page 3)	Module implementing the segmentation of an audio file using the package VMO.
<i>VMO_Score.petri_net</i> (page 4)	Module implementing the methods for the construction of a petri net from a segmentation analysis.
<i>VMO_Score.improviser</i> (page 6)	Module implementing methods for the off-line improvisation with an existing
<i>VMO_Score.utils</i> (page 7)	Module implementing some useful methods.

VMO_Score.segmentation

Module implementing the segmentation of an audio file using the package VMO.

Classes

<i>Region</i> (page 3)(min, max)	
<i>Section</i> (page 3)	Class representing a section of the segmentation
<i>Segmentation</i> (page 3)(audio_path)	Class representing a segmentation

```
class VMO_Score.segmentation.Region (min, max)
```

```
    max  
        Alias for field number 1
```

```
    min  
        Alias for field number 0
```

```
class VMO_Score.segmentation.Section  
    Class representing a section of the segmentation
```

class `VMO_Score.segmentation.Segmentation` (*audio_path*)

Class representing a segmentation

generate (*audio_path*, *sr=44100*, *hop_length=512*)

Segmentation of an audio recording using the Constant Q Transform (CQT)

Parameters

- **audio_path** (*str*) – A string representing the path of the audio file
- **sr** (*int*) – Sampling rate (default: 44100)
- **hop_length** (*int*) – Number of samples between successive CQT columns
- **(default** – 512.

oracle

VMO Oracle generated with the segmentation.

Returns Oracle

Return type VMO

output_png (*filename*)

Generate a PNG file of the segmentation

Parameters filename (*str*) – A string representing the path of the generated file

save_oracle (*filename*)

Save the oracle to disk

Parameters filename (*str*) – Filename to write the oracle

segmentation

List of sections generated by the segmentation.

Returns [Section]

VMO_Score.petri_net

Module implementing the methods for the construction of a petri net from a segmentation analysis.

Classes

PetriNet (page 5)([s, pnml])

Class that represents a Petri Net

Exceptions

ExpressionError (page 5)(arg)

Class representing an error raised during the conversion to Expressions.

exception `VMO_Score.petri_net.ExpressionError` (*arg*)

Class representing an error raised during the conversion to Expressions.

arg

str – String representing the information of the error.

class `VMO_Score.petri_net.PetriNet` (*s=None, pnml=None*)

Class that represents a Petri Net

data

Dictionary representing the regions belonging to each place.

Returns Dictionary where keys are petri net places and values are their segmentation regions

Return type dict

final_place

Returns the final place of the Petri Net.

Returns id of the final place

Return type str

from_pnml (*filepath*)

Load a Petri Net from a PNML file

Parameters *filepath* (*str*) – Path of the PNML file

from_segmentation (*segmentation*)

Build a Petri Net representing the segmentation of an audio file.

Parameters *segmentation* (`Segmentation` (page 3)) – Segmentation

initial_place

Returns the initial place of the Petri Net

Returns id of the initial place

Return type str

output_png (*filename*)

Save the Petri Net as a PNG file

Parameters *filename* (*str*) – String representing the path of the output file

pn

Object representing the Petri Net model.

Returns the Petri net

Return type `PetriNet` (page 5)

print_transitions ()

Print the information of the transitions of a Petri Net

Print the name, the minimum and maximum duration and the current time of all transitions of a Petri Net

to_json (*filename*)

Save the Petri Net to a json file readily by i-score

Parameters *filename* (*str*) – String representing the output file

to_pnml (*filename*)

Save the Petri Net as a PNML file

Parameters filename (*str*) – String representing the path of the output file

update_from_config (*filename*)

Update a Petri Net with a configuration file

Read a configuration file and update the transition parameters with the values specifies on it.

Parameters filename (*str*) – String representing the path of the configuration file

update_transition (*t_name, new_min, new_max, new_guard*)

Update a timed transition of a Petri Net

Update a timed transition with new minimum and maximum durations and a guard.

Parameters

- **t_name** (*str*) – Name of the transition to be changed
- **new_min** (*float*) – Minimum duration of the transition
- **new_max** (*float*) – Maximum duration of the transition
- **new_guard** (*Expression*) – Condition of the transition

VMO_Score.improviser

Module implementing methods for the off-line improvisation with an existing Petri net

Classes

<i>Improviser</i> (page 6)(<i>pn, path_conf</i>)	Class representing the off-line improviser
defaultdict	defaultdict(default_factory[, ...]) → dict with default factory

class VMO_Score.improviser.**Improviser** (*pn, path_conf*)

Class representing the off-line improviser

add_action (*addr, value*)

Add a user action to the Petri Net.

Add a new token to the place that represents the actions sent by the environment,

Parameters

- **addr** (*str*) – String representing the OSC address of the message
- **value** (*str*) – String representing the OSC value of the message

current_time

float – the amount of time that has elapsed

delete_action (*addr, value*)

Remove a user action from the Petri Net

Remove a token from the place that represents the actions sent by the environment.

Parameters

- **addr** (*str*) – String representing the OSC address of the message
- **value** (*str*) – String representing the OSC value of the message

fire_transition (*t_name*)

Fire a transition

Take randomly a mode of an enabled transition and fire it

Parameters **t_name** (*str*) – Name of the transition to be fired

get_enabled_transitions ()

Enabled transitions

Return a list of enabled transitions with the current marking

is_final_marking ()

Return if the current state of the Petri Net is a final state

Returns The Petri Net is in a final state

Return type bool

make_step ()

Make a step in a Petri Net

Take a logical step in the Petri Net. That is, an enabled transition is taken randomly and then it is fired.

next_time_unit ()

Increment the global clock and the transition time by one time-unit

pn

PetriNet – the Petri net for improvisation

restart ()

Restart to the initial marking of the Petri Net

VMO_Score.utils

Module implementing some useful methods.

Functions

<i>generate_configuration</i> (page 7)(filepath, pn)	Generate a configuration file for the Petri Net.
<i>load_configuration</i> (page 8)(filepath)	Load a configuration file.
<i>load_shelve</i> (page 8)(filepath, field)	Load a file into an object.
<i>save_shelf</i> (page 8)(filepath, field, obj)	Save an object into a file.

VMO_Score.utils.**generate_configuration** (*filepath*, *pn*)

Generate a configuration file for the Petri Net.

Parameters

- **filepath** (*str*) – String representing the path of the output
- **pn** (*PetriNet* (page 5)) – A Petri Net

`VMO_Score.utils.load_configuration(filepath)`

Load a configuration file.

Parameters **filepath** – String representing the path of the configuration file

Returns Dictionary representing the configuration file

Return type dict

`VMO_Score.utils.load_shelve(filepath, field)`

Load a file into an object.

Parameters

- **filepath** (*str*) – String representing the path of the file
- **field** (*str*) – identifier of the object

Returns Dictionary representing the object

Return type dict

`VMO_Score.utils.save_shelf(filepath, field, obj)`

Save an object into a file.

Parameters

- **filepath** (*str*) – Path of the file
- **field** (*str*) – Id of the object in the file
- **obj** (*object*) – Object to be saved

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

V

VMO_Score.improviser, 6
VMO_Score.petri_net, 4
VMO_Score.segmentation, 3
VMO_Score.utils, 7

A

add_action() (VMO_Score.improviser.Improviser method), 6
 arg (VMO_Score.petri_net.ExpressionError attribute), 5

C

current_time (VMO_Score.improviser.Improviser attribute), 6

D

data (VMO_Score.petri_net.PetriNet attribute), 5
 delete_action() (VMO_Score.improviser.Improviser method), 6

E

ExpressionError, 5

F

final_place (VMO_Score.petri_net.PetriNet attribute), 5
 fire_transition() (VMO_Score.improviser.Improviser method), 7
 from_pnml() (VMO_Score.petri_net.PetriNet method), 5
 from_segmentation() (VMO_Score.petri_net.PetriNet method), 5

G

generate() (VMO_Score.segmentation.Segmentation method), 4
 generate_configuration() (in module VMO_Score.utils), 7
 get_enabled_transitions() (VMO_Score.improviser.Improviser method), 7

I

Improviser (class in VMO_Score.improviser), 6

initial_place (VMO_Score.petri_net.PetriNet attribute), 5
 is_final_marking() (VMO_Score.improviser.Improviser method), 7

L

load_configuration() (in module VMO_Score.utils), 8
 load_shelve() (in module VMO_Score.utils), 8

M

make_step() (VMO_Score.improviser.Improviser method), 7
 max (VMO_Score.segmentation.Region attribute), 3
 min (VMO_Score.segmentation.Region attribute), 3

N

next_time_unit() (VMO_Score.improviser.Improviser method), 7

O

oracle (VMO_Score.segmentation.Segmentation attribute), 4
 output_png() (VMO_Score.petri_net.PetriNet method), 5
 output_png() (VMO_Score.segmentation.Segmentation method), 4

P

PetriNet (class in VMO_Score.petri_net), 5
 pn (VMO_Score.improviser.Improviser attribute), 7
 pn (VMO_Score.petri_net.PetriNet attribute), 5
 print_transitions() (VMO_Score.petri_net.PetriNet method), 5

R

Region (class in VMO_Score.segmentation), 3
restart() (VMO_Score.improviser.Improviser
method), 7

S

save_oracle() (VMO_Score.segmentation.Segmentation
method), 4
save_shelf() (in module VMO_Score.utils), 8
Section (class in VMO_Score.segmentation), 3
Segmentation (class in VMO_Score.segmentation), 3
segmentation (VMO_Score.segmentation.Segmentation
attribute), 4

T

to_json() (VMO_Score.petri_net.PetriNet
method), 5
to_pnml() (VMO_Score.petri_net.PetriNet
method), 5

U

update_from_config()
(VMO_Score.petri_net.PetriNet
method), 6
update_transition()
(VMO_Score.petri_net.PetriNet
method), 6

V

VMO_Score.improviser (module), 6
VMO_Score.petri_net (module), 4
VMO_Score.segmentation (module), 3
VMO_Score.utils (module), 7