The `optprob` package*

Yudai NAKATA

August 7, 2016

Abstract

`optprob` is a package that offers you useful features for writing optimization problems in a structured manner. The package also provides highly customizable options for the appearance of the optimization problems.

Contents

1 Introduction

1.1 Motivation

In order to illustrate optimization problems, one can utilize `alignat` environment from the `amsmath` package. A simple example like this (assuming `amsmath` is loaded),

*https://github.com/yudai-nkt/optprob

```
\DeclareMathOperator*\maximize{maximize}
\begin{alignat}{2}
    & \mathord{\maximize_{x_1,x_2\in\mathbf{R}}} & \quad & c_1x_1+c_2x_2 \\
    & \text{subject to} & & a_{11}x_1+a_{12}x_2 \leq b_1 \\
    & & & a_{21}x_1+a_{22}x_2 \leq b_2.
\end{alignat}
```

will produce

$$\begin{alignat}{2}
&\underset{x_1,x_2\in\mathbf{R}}{\text{maximize}} && c_1x_1+c_2x_2 && (1)\\
&\text{subject to} && a_{11}x_1+a_{12}x_2 \leq b_1 && (2)\\
& && a_{21}x_1+a_{22}x_2 \leq b_2. && (3)
\end{alignat}$$

One can also use other `align`-variant environments or `array` environment.

Each method has its pros and cons, but none of them are very friendly when it comes to denoting optimization problem. Writing disgusting numbers of ampersands in every line for the proper alignment is really annoying (to me at least) and could sometimes look ugly if we only input inadequate ampersands. Also, it is difficult to intuitively grasp the role of each formulae at a glance upon the source code if we put them manually.

1.2 Objective of this package

The `optprob` package provides an elegant and semantic syntax for optimization problems. With the aid of this package, the optimization problem above can be marked up as follows.

```
\begin{optimize}{max}
    \objfunc[x_1,x_2\in\mathbf{R}]{c_1x_1+c_2x_2}
    \addconstraint{a_{11}x_1+a_{12}x_2 \leq b_1}
    \addconstraint{a_{21}x_1+a_{22}x_2 \leq b_2.}
\end{optimize}
```

You are free from the concern about formatting the problems. You can also easily switch the spacing or other settings via $\langle key \rangle = \langle value \rangle$ syntax. Semantically-named macros enhance your code's readability compared to the bare LaTeX expression.

2 Requirements

The `optprob` package is supported on the following engines and format.

- TeX engine: TeX, pdfTeX, XeTeX, LuaTeX, pTeX and upTeX
- TeX format: LaTeX 2_ε (plain TeX and ConTeXt are not supported.)

This package requires `mathtools` and `pgfopts` to make use of it, and listed below are necessary packages to typeset this documentation besides the prerequisite packages:

- `cleveref`
- `fontenc`
- `geometry`
- `hologo`
- `lmodern`
- `ltxdockit`
- `minted`
- `multicol`

All the packages above are bundled with recent TeX Live by default. If any of them are not installed in your TeX system, you can download them via CTAN.

3 Installation

The `optprob` package is currently available only at GitHub. Download the TDS (TeX Directory Structure) archive `optprob.tds.zip` from the GitHub Releases, and move each file to either `TEXMFLOCAL` or `TEXMFHOME` keeping the directory hierarchy. Here is a quick script[1] for UNIX-like OS users:

```
curl -O https://github.com/yudai-nkt/optprob/releases/download/v0.2.0/optprob.tds.zip
unzip optprob.tds.zip -d $(kpsewhich -var-value TEXMFHOME)
```

Make sure to run `mktexlsr` if necessary.

4 Usage

4.1 Package loading

Just put this line in your preamble:

```
\usepackage[⟨options⟩]{optprob}
```

For the time being, this package has the following options available.

`ignorecfg` This option tells the package to skip reading `optprob.cfg`. Details about this file and option are in the Section 4.3.

`mathtools` This option is specified with ⟨*key*⟩ = ⟨*value*⟩ syntax. If ⟨*mathtools options*⟩ is set to the value of `mathtools` key, it is passed to `mathtools` package's option. You need to enclose ⟨*mathtools options*⟩ by curly braces if more than one options are passed.

4.2 User level commands

`\argmax`

This command outputs "arg max", which denotes arguments of the maximum.

`\argmin`

This command is the minimum counterpart of `\argmax`.

[1] Use `TEXMFLOCAL` instead if you install there.

`\begin{optimize}`[⟨*layout formatting*⟩]{⟨*operation type*⟩}

`\end{optimize}`

Either `max` or `min` can be used for ⟨*operation type*⟩ according to the problem you want to represent. You can specify the design of the environment in the ⟨*layout formatting*⟩ option. Each setting can be given in ⟨*key*⟩ = ⟨*value*⟩ syntax as follows.

`abbrev=true, false` default: `false`

This key tells whether or not to abbreviate the terms such as maximize, minimize and subject to.

`showeqnum=true, false` default: `true`

This key tells whether or not to show the equation numbers in each line.

`space=`⟨*dimension*⟩ default: `1em`

This key sets the space between two columns.

`\begin{maximize}`[⟨*layout formatting*⟩]

`\end{maximize}`

This environment is equivalent to the `optimize` environment with the mandatory argument set to `max`.

`\begin{minimize}`[⟨*layout formatting*⟩]

`\end{minimize}`

This environment is equivalent to the `optimize` environment with the mandatory argument set to `min`. Note that this and `maximize` environments are defined only if `\minimize` and `\maximize` are not defined in the preamble respectively.

Within these environments above, the following macros are locally defined:

`\objfunc`[⟨*variables*⟩]{⟨*objective function*⟩}

This command sets the objective function to be maximized or minimized and denotes the variables over which the objective function is optimized. The default value for optional argument is an empty string. You *must* use this macro once per one environment.

`\addconstraint`{⟨*constraint*⟩}

This command adds constraints of the problem. You can use this macro as many times as necessary.

Let me go back to explanation of global commands.

`\maximizeinline`[⟨*options*⟩]{⟨*objective function*⟩}{⟨*constraint*⟩}

This command denotes an inline maximization problems.

`abbrev=true, false` default: `false`

This key is the same as the corresponding option of `optimize` environment.

`variable=`⟨*variable*⟩ default: `(no value)`

This key sets the variables over which the objective function is optimized.

`\maximizeinline*[`⟨*options*⟩`]{`⟨*objective function*⟩`}`

Starred `\maximizeinline` is used when explicit representation of constraint is unnecessary.

`\minimizeinline[`⟨*options*⟩`]{`⟨*objective function*⟩`}{`⟨*constraint*⟩`}`

This command is the minimum counterpart of `\maximizeinline`.

`\minimizeinline*[`⟨*options*⟩`]{`⟨*objective function*⟩`}`

This command is the minimum counterpart of `\maximizeinline*`.

4.3 The `optprob.cfg` file

If your TEX system finds a file named `optprob.cfg` somewhere TEX can search, that file is loaded automatically. The primary purpose of `optprob.cfg` is to store your frequent customizations so that you do not have to spend time and energy writing repetitive preamble every time you make new material.

You sometimes do not want the package to load your `optprob.cfg` even if it exists. In such cases, use `ignorecfg` option, which will disable the autoload. This option does nothing when `optprob.cfg` is not found.

5 Acknowledgements

The author is thankful to those who made and/or have been mantaining the packages on which `optprob` has dependency.

6 License

7 Changelog

v0.2.0 (July 5, 2016)

- Add an option for switching the appearance of equation numbers.
- Change the way of specifying variables in `optimize` environment and friends.

v0.1.1 (July 5, 2016)

- Fix the incompatibility with `minipage` environment.

v0.1.0 (May 25, 2016)

- Fisrt publication.