PRL/APS and arXiv submission guide

P. Fowler-Wright
pfw1@st-andrews.ac.uk

2023-03-21

Contents

1 What this document covers 1

2 Writing the manuscript 2

3 Preparation 2

4 Generating manuscript (paper) files for PRL 3

5 Generating SM (supp) for PRL 3

6 arXiv submission 4

1 What this document covers

This document describes the process of submitting a manuscript to Physical Review Letters while also making a copy available on `https://arxiv.org/`. The starting point is that you've written a manuscript (the Letter) plus additional Supplementary Material (SM) in a *single*[1] LaTeX document.

The main contribution of this work is the helper scripts `parse_main_aux.py` and `select_links.py` which may be used to generate SM with references to both items in the main text (Letter) *and* the SM itself, but with active hyperlinks only in the case the item is in the SM, while at the same time preserving the correct bibliography structure (the reference list of the Letter includes articles cited in both the Letter and the SM).

Note we submit our manuscript as a `.tex` file with a compiled bibliography (`.bbl`) file: both APS and arXiv generate a `.pdf` from the raw files. On the other hand, APS (at least PRL) accepts SM as a separate `.pdf`, which you must generate locally.

Submission to other APS journals is likey to be very similar, and may not have the complication of having separate SM (e.g. included as Appendices).

[1] Writing SM in a separate document from the outset may be feasible, but you would need to figure out how to add any citations to the main bibliography (with the correct) numbers and make consistent references to figures, equations etc. in the main text.

You shouldn't use this document if... You want a simple solution and aren't bothered by not having working hyperlinks for SM items. In that case, the easiest thing to do is to compile the entire document (resetting the page numbering for the SM perhaps) and simply extract the SM pages to create the SM `.pdf`. To generate the Letter `.tex`, simply remove all the SM text (you may need to make dummy `cites` to ensure the Letter bibliography contains all references).

Todos

- Guide to using the APS submission widget including section, header choices
- Guide to using the arXiv submission widget, plus considerations of submission timing, authorship
- updating an arXiv submission

2 Writing the manuscript

See `prl_template.tex` for a basic template that may be used to create a `revtex-4.2` (the APS/AIP article class) document containing both manuscript and SM. Whilst writing the manuscript you will want to bear in mind the Physical Review style guide and word count limits. There is also an author's guide for the `revtex` class.

3 Preparation

Suppose we are in a directory containing our paper files:

```
paper_files/
    - main.tex
    - refs.bib
    - fig1.pdf
    - fig2.pdf
    - figSM1.pdf
    - figSM2.pdf
```

where `main.tex` is the manuscript plus SM, `refs.bib` the bibliography used by the main file (`bibliographyrefs.bib`) and `fig1.pdf`, `fig2.pdf` (`figSM1.pdf`, `figSM2.pdf`) figures used in the manuscript (SM). Note when submitting to PRL and arXiv all files must reside in the same (top-level) directory, so you can't e.g. use a `figures/` subdirectory containing all four figures.

Since we will submit the `.tex` directly, it is a good idea to clean up `main.tex` by removing any comments or unused code (this file will be publicly accessible via 'Download Source' on arXiv). Once you think your manuscript content is ready to make a submission (or re-submission), compile `main.tex` as you normally would i.e. running both a LaTeX compiler and BIBTeX program. *Check the output* `main.pdf` in case anything broke whilst you were removing commens etc. (we will be checking the output a lot).

From the files produced we will need `main.aux` and `main.bbl`, in addition to `main.tex` and any figure/data files required by `main.tex` (all other files can be left as they are). We will also need the scripts `parse_main_aux.py` and `select_links.py` to generate the SM. Create two new subdirectories `paper` and `supp` with

```
paper_files/paper/
    - main.tex
    - main.bbl
    - fig1.pdf, fig2.pdf # ANY figures/data used in main text (Letter)
paper_files/supp/
    - main.tex
    - main.aux
    - figSM1.pdf, figSM2.pdf # ANY figures/data used in SM
- parse_main_aux.py, select_links.py # helper scripts
```

4 Generating manuscript (paper) files for PRL

Move into `paper_files/paper` (`cd paper`) and edit `main.tex`:

- Remove *all* SM text (content *between* `\biliography{refs.bib}` and `\end{document}`)
- Replace `\biliography{refs.bib}` with `\input{main.bib}`
- Remove the `\}` arXiv command

The document should now be compiled *twice* using LaTeX only i.e. not BIBTeX. Since most editors will try to run BIBTeX automatically, I recommend doing this from the command line:

```
$ pdflatex main.tex && pdflatex main.tex
```

Two compilations are needed to get hyperlinks working. *Check the output* `main.pdf`. This should have the Letter with all hyperlinks (refs/cites) present and functional.

`main.tex`, `main.bbl` and any Letter figures/data are now ready to be uploaded to the APS submission server (todo: details on submission server UI). The following command creates a tarball `upload.tar` that can be uploaded directly:

```
$ tar -cvf upload.tar main.tex main.bbl fig1.pdf fig2.pdf
```

Do *not* include `main.pdf` or other auxiliary files produced when you ran `pdflatex`.

5 Generating SM (supp) for PRL

Move into `paper_files/supp` and edit `main.tex`:

- Remove *all* Letter text (content *after* `\begin{document}` up to and including `\bibliography{refs.bib}\clearpage`)

Now run `parse_main_aux.py main.aux` to produce `out.aux`. This removes any link that isn't a `\citation`, `\newlabel` or `\bibcite` commands (it also removes all commands corresponding to SM labels/citations). This is done to preserve references in the final SM `.pdf`. This list of commands is almost certainly incomplete, so you may find we need to add others (let me know). Once we think this ran successfully (or would like to test it), rename `out.tex` to `labels.aux`.

Next run `select_links.py main.tex` to produce `out.tex`. This script finds commands e.g. `\cite{kubo1962}`, `\eqref{eq:1}` that reference labels from the Letter, and wraps these references with a command `\nolink` (defined below) which suppresses the creation of a clickable link for these elements. The point is in the final SM we want to be click references to items (equations,

figures etc.) in the SM itself, but not items in the Letter. This is done using regular expressions (Python `re`), and will almost certainly not work perfectly the first time you use it given the variety of commands available to references in a LaTeX document: `\cite{}`, `\Cite{}`, `\ref{}`, `\eqref{}`, `\cref{}`, `\Cref{}`, should certainly all work, but get in touch with me to fix the script for your use case. Once things look to be working (or we want to test whether they are), rename `out.tex` to `supp.tex`.

Finally, edit `supp.tex` and immediately *above* `\begin{document}` add

```
\input{labels.aux} % file generated by parse_main_aux.py
\newcommand*{\nolink}[1]{%
  {\protect\NoHyper#1\protect\endNoHyper}%
}
```

Do a double LaTeX compilation of `supp.tex`:

```
$  pdflatex supp.tex && pdflatex supp.tex
```

If everything worked, `supp.pdf` should contain SM with all references present, but only those referencing items in the SM itself featuring an active link (you may chose to disable the link colour by passing `[colorlinks=false]` or `[hidelinks]` options to the `hyperref` package in the preamble. `supp.pdf` can now be uploaded as it's own document using the APS submission widget (alongside the files uploaded in the previous section).

6 arXiv submission

This process is far simpler as our main file has the format we want for arXiv already. Create a directory `paper_files/arxiv` and copy `main.tex`, `main.bbl` and *all* (Letter plus SM) figures/data files there. Now simply create a single tarball which can be directly uploaded to arXiv:

```
$ cd arxiv
$ tar -cv upload.tar *
```

A more detailed guide is available at https://trevorcampbell.me/html/arxiv.html.