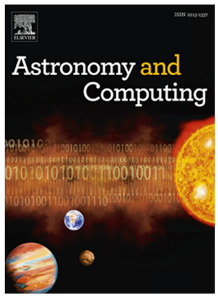
# Accepted Manuscript




Minimal re-computation for exploratory data analysis in astronomy

B. Nikolic, D. Small, M. Kettenis

PII: S2213-1337(18)30024-6
DOI: https://doi.org/10.1016/j.ascom.2018.09.003
Reference: ASCOM 248

To appear in: *Astronomy and Computing*

Received date : 28 February 2018
Accepted date : 4 September 2018

Please cite this article as: Nikolic B., et al., Minimal re-computation for exploratory data analysis in astronomy. *Astronomy and Computing* (2018), https://doi.org/10.1016/j.ascom.2018.09.003

# Minimal Re-computation for Exploratory Data Analysis in Astronomy

Bojan Nikolic

*Cavendish Laboratory, University of Cambridge, United Kingdom*

Des Small and Mark Kettenis

*Joint Institute for VLBI ERIC, Dwingeloo, The Netherlands*

**Abstract**

We present a technique to automatically minimise the re-computation when a data analysis program is iteratively changed, or added to, as is often the case in exploratory data analysis in astronomy. A typical example is flagging and calibration of demanding or unusual observations where visual inspection suggests improvement to the processing strategy. The technique is based on memoization and referentially transparent tasks. We describe the implementation of this technique for the CASA radio astronomy data reduction package. We also propose a technique for optimising efficiency of storage of memoized intermediate data products using copy-on-write and block level de-duplication and measure their practical efficiency. We find the minimal recomputation technique improves the efficiency of data analysis while reducing the possibility for user error and improving the reproducibility of the final result. It also aids exploratory data analysis on batch-schedule cluster computer systems.

*Keywords:* methods: data analysis; functional languages

## 1. Introduction

Notwithstanding the notable successes in automating the reduction and analysis of data from radio telescopes, the traditional astronomer-driven data reduction is still common. This typically takes the form of exploratory, iterative, data reduction where visual inspection of intermediate or final data products is used to adjust, or add to, the data processing program. Each adjustment is typically small and impacts only a subset of the processing as a whole; however in current systems there is no way automatically re-run only this subset. Instead the user has the choice to either re-run the whole program, which can take minutes, hours or even days; or to manually do the sub-selection of the part of the program which needs to be re-run and risk introducing errors.

The situation can be easily be improved as we show below, by thin wrappers around existing data reduction software systems and applying techniques used in