Monthly Notices
of the
ROYAL ASTRONOMICAL SOCIETY

Mon. Not. R. Astron. Soc. **422,** 1812–1824 (2012) doi:10.1111/j.1365-2966.2012.20768.x

Compact continuum source finding for next generation radio surveys

P. J. Hancock,[1,2]⋆ T. Murphy,[1,2,3] B. M. Gaensler,[1,2] A. Hopkins[2,4] and J. R. Curran[3]

[1] *Sydney Institute for Astronomy, School of Physics A29, The University of Sydney, NSW 2006, Australia*
[2] *ARC Centre of Excellence for All-Sky Astrophysics (CAASTRO), The University of Sydney, NSW 2006, Australia*
[3] *School of Information Technologies, The University of Sydney, NSW 2006, Australia*
[4] *Australian Astronomical Observatory, PO Box 296, Epping, NSW 1710, Australia*

Accepted 2012 February 19. Received 2012 February 17; in original form 2011 October 5

ABSTRACT

We present a detailed analysis of four of the most widely used radio source-finding packages in radio astronomy, and a program being developed for the Australian Square Kilometer Array Pathfinder telescope. The four packages: SEXTRACTOR, SFIND, IMSAD and SELAVY are shown to produce source catalogues with high completeness and reliability. In this paper we analyse the small fraction (∼1 per cent) of cases in which these packages do not perform well. This small fraction of sources will be of concern for the next generation of radio surveys which will produce many thousands of sources on a daily basis, in particular for blind radio transients surveys. From our analysis we identify the ways in which the underlying source-finding algorithms fail. We demonstrate a new source-finding algorithm AEGEAN, based on the application of a Laplacian kernel, which can avoid these problems and can produce complete and reliable source catalogues for the next generation of radio surveys.

Key words: techniques: image processing – catalogues – surveys.

1 INTRODUCTION

Source finding in radio astronomy is the process of finding and characterizing objects in radio images. The properties of these objects are then extracted from the image to form a survey catalogue. The aim of large-scale radio imaging surveys is to provide an unbiased census of the radio sky, and hence the ideal source finder is both complete (finds all sources present in the image) and reliable (all sources found and extracted are real).

Most of the standard source-finding algorithms that have been developed over the last few decades are highly complete and reliable, missing only a small fraction of sources. These problem cases are generally dealt with in pre- or post-processing, or manually corrected in the source catalogue.

Next generation radio surveys such as the Evolutionary Map of the Universe (EMU; Norris et al. 2011) and the Australian Square Kilometer Array Pathfinder (ASKAP) Survey for Variables and Slow Transients (VAST; Chatterjee et al. 2010) planned for the ASKAP (Johnston et al. 2008) telescope will produce large area, sensitive maps of the sky at high cadence, resulting in many times more data than previous surveys. Data processing will need to be fully automated, with limited scope for manual intervention and correction. Hence the small number of missing or incorrectly identified sources produced by current source finders will pose a substantial problem. In particular, in blind surveys for radio transients, missed sources and false positives in an epoch will cause the transient detection algorithms to trigger on false ‘events’. VAST will need to extract thousands of sources from survey images at a cadence of ∼5 s. A source-finding algorithm which is 99 per cent complete and reliable at a signal-to-noise ratio (SNR) of 5 will be producing ∼10 000 false sources, and missing ∼10 000 real sources per day. Whilst it is possible to remove false sources from a catalogue, the missing real sources are lost forever. The large data rates of telescopes like ASKAP will make it impossible to store each observation, and thus no reprocessing of the data will be possible.

The way in which a source-finding algorithm fails to detect a real source is often assumed to be related to noise, and that it is random. In this paper we test this assumption and show that whilst many sources are missed due to random noise related effects, there is also a component that is deterministic and related to the underlying algorithm. By analysing the source-finding algorithms and their modes of failure we identify ways in which the algorithms could be improved and use this knowledge to build an algorithm which can produce catalogues that are more complete and more reliable than those currently available.

In this paper we discuss the problem of source finding for Stokes *I* continuum radio emission in the context of next generation radio imaging surveys. We will not deal directly with the additional complications introduced by spectral line data, polarization data or extended sources and diffuse emission. The focus of this paper is on upcoming surveys for ASKAP, however, the results will be equally applicable to future radio surveys on other Square Kilometer Array (SKA) pathfinder instruments such as the Murchison Widefield

⋆E-mail: Paul.Hancock@sydney.edu.au

Array (MWA; Lonsdale et al. 2009) and SKA Molonglo Prototype (SKAMP; Adams, Bunton & Kesteven 2004), and of course the SKA itself.

In Section 2 we outline the main approaches to source finding in radio astronomy, and then in Section 3 describe four of most widely used source-finding packages. Section 3.7 gives some examples of the way that source-finding packages are used to create catalogues for large surveys and transient studies. Section 4 describes the test data that were used in the analysis of the source-finding algorithms, and Section 5 describes the evaluation process. The instances in which the source-finding algorithms fail to find or properly characterize sources are described in Section 6. Section 7 describes a new source-finding algorithm, AEGEAN, which has been designed to overcome many of the problems suffered by existing source-finding packages. We summarize our conclusion in Section 8.

2 SOURCE FINDING IN RADIO ASTRONOMY

In a broad sense, source finding in radio images involves finding pixels that contain information about an astronomical source. Most approaches to source finding in radio astronomy follow a similar method: (i) background estimation and subtraction; (ii) source identification; (iii) source characterization and (iv) cataloguing. In this section we outline the standard method taken in each of these steps.

In the discussion that follows we consider a source to be a signal of astronomical importance that can be well modelled by an elliptical Gaussian. By this definition a radio galaxy with a typical core/jet morphology would be made up of three sources, one for the jet and each of the lobes. The grouping of multiple sources into a single object of interest (like a core/jet radio galaxy) is not in the realm of source finding or classification as it relies on contextual information to make such an association.

2.1 Background estimation and subtraction

The first step in source finding is determining which parts of the image belong to sources and which belong to the background (e.g. Huynh et al. 2011). The most common way in which this separation is achieved is to set a flux threshold that divides pixels in to background or source pixels. This process is referred to as thresholding.

A straightforward case would involve a background that is dominated by thermal noise, which is without structure and is constant across the entire image. In such a case a single threshold value can be chosen that will result in all sources above that threshold being detected, and some small number of false detections. A varying background can be accounted for by calculating the mean and rms noise in local subregions, which is then used to normalize the image before applying a uniform threshold in SNR. The selection of a threshold limit is often a balance between detecting as many real sources as possible and minimizing the number of false detections. Typically a 5σ threshold limit is used in a blind survey, with higher or lower limits chosen for larger or smaller regions of sky.

False detection rate (FDR) analysis (Hopkins et al. 2002) determines the threshold limit that will result in a number of falsely detected pixels that is lower than some user defined limit.

In cases where the background has structure, an image filter must be used to remove the background structure before the source-finding stage. The way in which the background structure is removed depends on the cause and type of structure that is present. A common example is diffuse emission in the galactic plane, with compact sources embedded within. A discussion of background filtering techniques is beyond the scope of this paper, and in our analysis we assume the images have been pre-processed and are free of background structure. For an evaluation of background estimation see Huynh et al. (2011).

2.2 Source identification

Source identification is the process by which pixels that are above a given threshold are grouped into contiguous groups called islands. Each island corresponds to one or more sources of interest. The process of *finding* sources is complete at this stage. The format of the catalogue is just a list of pixels that belong to each of the islands, which is not of general astronomical utility. Source characterization is required to convert these islands of pixels into a more useful form.

2.3 Source characterization

Source characterization involves measuring the properties of each source, for example the total flux and angular size. The best source characterization method is strongly dependent on the nature of the sources that are to be studied. Point sources, by definition, have the shape of the point spread function (PSF) of an image, making the PSF shape important in the characterization process. Images that are produced from radio synthesis observations have been deconvolved and the complicated PSF of the instrument has been replaced with an appropriately scaled Gaussian. Observations with sufficient u, v coverage will do not need to be deconvolved as they have a PSF that is already nearly a Gaussian. In either case, compact sources will appear as Gaussian, and so an island of pixels can be characterized by a set of Gaussian components.

In lower resolution radio surveys such as the NRAO VLA Sky Survey (NVSS, 45 arcsec^2; Condon et al. 1998) and Sydney University Molonglo Sky Survey (SUMSS, $45 \times 45\mathrm{cosec}|\delta|$ arcsec^2; Mauch et al. 2003) a majority of objects are unresolved and can be characterized by a single Gaussian. However, in higher resolution surveys such as Faint Images of the Radio Sky at Twenty-Centimeters (FIRST, 5 arcsec^2; Becker, White & Helfand 1995) a significant fraction of the sources are partially extended or have multiple components, and so multiple Gaussians are required to represent them.

Fitting a number of Gaussian components to an island of pixels is straightforward (Condon 1997), but is highly sensitive to the choice of initial parameters. Gaussian fitting can converge to unrealistic or non-optimal parameters due to the many local minima in the difference function. Effective multiple Gaussian fitting requires two things: an intelligent estimate of the starting parameters, and sensible constraints on these parameters. None of the widely used source-finding packages has an algorithm for robustly estimating initial parameters for a multiple Gaussian fit.

Two approaches have been developed which try to address the difficulty of obtaining accurate initial parameters for multiple Gaussian fitting: de-blending and iterative fitting. A de-blending-based approach breaks an island into multiple sub-islands, each of which is fit with a single component. In an iterative fitting approach, the difference between the image data and the fitted model (the fitting residual) is evaluated in order to determine whether an extra component is required. This analysis will repeat until an acceptable fit is achieved, or a limit on the number of components has been reached. De-blending and iterative fitting are both susceptible to source fragmentation, whereby a single true source is erroneously represented by multiple components.

Once each island of pixels has been characterized the fitting parameters are catalogued.

2.4 Cataloguing

The final stage in source finding is extracting the source parameters and forming a catalogue of objects in the field.

A catalogue should contain an appropriate listing of every parameter that was fit, along with the associated uncertainties. In addition to the fitted parameters, a source-finding algorithm should report instances where the source characterization stage was inadequate or failed. By reporting sources that were not well fit, a catalogue can remain complete despite having measured some source parameters incorrectly. Poorly fit sources can easily be remeasured, whereas excluded sources are missed forever. If it is not possible to construct a reasonable facsimile of the true sky using only the information provided in the source catalogue then the source-finding process has not been successful.

3 SOURCE-FINDING PACKAGES AND THEIR ALGORITHMS

Most of the major source-finding packages in astronomy are based on a few common algorithms. In this section we outline the features of these packages.

Source-finding packages that rely on wavelet analysis were not considered in this work as none of the most widely used source-finding packages relies on wavelet analysis.

3.1 SEXTRACTOR

SEXTRACTOR (SE; Bertin & Arnouts 1996) was developed for use on optical images from scanned plates. The speed and ease of use of SEXTRACTOR has made it a popular choice for radio astronomy despite its optical astronomy origins. SEXTRACTOR is a stand alone package for UNIX-like operating systems.

The source finding and characterization process that SEXTRACTOR follows can be modified via an extensive parameter file. For this work the following parameters were used:

DETECT_MINAREA	5
THRESH_TYPE	ABSOLUTE
DETECT_THRESH	125e-6
ANALYSIS_THRESH	75e-6
MASK_TYPE	CORRECT
BACK_SIZE	400
BACK_FILTERSIZE	3

The first four parameters instruct SEXTRACTOR to detect all sources with a peak pixel brighter than $5\sigma = 125\,\mu\mathrm{Jy\,beam^{-1}}$. The source characterization is then carried out on islands of pixels that are brighter than $3\sigma = 75\,\mu\mathrm{Jy\,beam^{-1}}$ that contain at least 5 pixels. The final three parameters ensure that the measured flux of a source is corrected for the effects of nearby sources, and that the background is estimated using a box of 3 × 400 pixels on a side. This large background size results in a background that is less than 1 μJy for each of the tested images. The parameters DEBLEND_NTHRESH and DEBLEND_MINCONT are used by SEXTRACTOR in the source characterization stage, when deciding how many components are contained within an island of pixels. The ability of SEXTRACTOR to characterize sources was found to be insensitive to these parameters for the simulated images used in this work.

3.2 IMSAD

Image Search and Destroy (IMSAD) is an image-based source-finding algorithm in MIRIAD (Sault, Teuben & Wright 1995). The threshold is user specified either as an absolute flux level or as a SNR with the background noise determined from a histogram of pixel values. Only pixels that are brighter than the threshold are used in the fitting process. For the analysis presented in this work we specify a threshold of $5\sigma = 125\,\mu\mathrm{Jy\,beam^{-1}}$. IMSAD performs a single Gaussian fit to each island of pixels.

3.3 SELAVY

SELAVY is the source-finding package that is being developed by ASKAPsoft as part of the data processing pipeline for ASKAP. SELAVY is a source-finding package that is able to work with spectral cubes and continuum images, and includes a number of different algorithms and approaches to source finding. SELAVY is related to the publicly available DUCHAMP software (Whiting 2012).[1] SELAVY is a version of the DUCHAMP software that has been integrated into the ASKAPsoft architecture to run on a highly parallel system with distributed resources. In the context of compact continuum source finding the only difference between SELAVY and DUCHAMP is that SELAVY is able to parametrize and island of pixels with multiple Gaussian components. SELAVY was given a threshold of $5\sigma = 125\,\mu\mathrm{Jy\,beam^{-1}}$ for source detection.

3.4 SFIND

SFIND (Hopkins et al. 2002) is implemented in MIRIAD and uses FDR analysis to set the detection threshold. Source characterization is performed using the same Gaussian fitting subroutine as that use by the MIRIAD task IMFIT.

A varying background is calculated by SFIND by dividing the image into subregions of (user defined) size, and measuring the mean and rms of each region. In an image which contains a high density of sources, or subregions which contain a particularly bright or extended source, the calculated mean and rms will be contaminated by the sources. For subregions where this occurs the result is a mean and rms value that is significantly different from the adjacent subregions, which can cause sources on the boundaries to be normalized such that their shape and flux distribution are not preserved. For an image which is constructed to have a zero mean and constant rms, these contamination effects can be largely removed by setting the size of the subregions to be larger than the given image.

The rejection of sources which fail to be fit with a Gaussian rejects many instances of sources that have very few pixels. This has the effect of further decreasing the FDR for the catalogue, since a false positive source needs to have many contiguous false positive pixels in order to be fit properly.

In this work we selected the subregions to be larger than the given image, and adjusted the FDR parameter until the automatically selected threshold was at $5\sigma = 125\,\mu\mathrm{Jy\,beam^{-1}}$.

3.5 FLOODFILL

FLOODFILL is an algorithm which performs the second stage of source finding, separating the foreground from the background pixels, and grouping them into islands that are then passed on to

[1] http://www.atnf.csiro.au/people/Matthew.Whiting/Duchamp

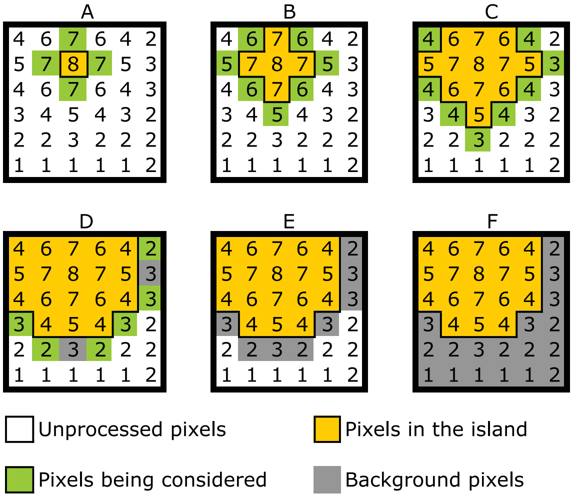


Figure 1. A demonstration of the operation of the FloodFill algorithm. The small 'image' has pixel values that are in units of the background noise. The seed threshold is $\sigma_{\rm s} = 5$ and the flood threshold $\sigma_{\rm f} = 4$. The orange pixels are those that have been assigned to an island, green pixels are those that are being considered, grey pixels have been rejected and white pixels are yet to be considered. In panel (A) the brightest pixels is used to seed an island. Pixels adjacent to the island are then inspected (coloured green). In panel (B) the pixels under consideration are brighter than the flooding threshold and are added to the island. The process is repeated in panels (C) and (D). In panel (E) there are no more pixels adjacent to the island that have not been inspected. In panel (F) all pixels have either been assigned to an island (orange) or are labelled as background (grey).

the source characterization stage. We describe FloodFill as implemented in the new source-finding algorithm, AEGEAN, which is described in Section 7. Although used by Murphy et al. (2007), the details of the algorithm have not been described in the astronomical literature (although see Roerdink & Meijster 2001).

FloodFill takes an image and two thresholds ($\sigma_{\rm s}$ and $\sigma_{\rm f}$, with $\sigma_{\rm s} \geq \sigma_{\rm f}$). Pixels that are above the seed threshold $\sigma_{\rm s}$ are used to seed an island, whilst pixels that are above the flood threshold $\sigma_{\rm f}$ are used to grow an island. Given a single pixel above $\sigma_{\rm s}$, FloodFill considers all the adjacent pixels. Adjacent pixels that are above $\sigma_{\rm f}$ are added to the island and pixels adjacent to these are then considered. This iterative process is continued until all adjacent pixels have been considered. The operation of FloodFill is demonstrated on a simplistic 'image' in Fig. 1. In panel (A) the brightest pixel in the image has been chosen to seed the island, and is coloured yellow. The adjacent pixels are coloured cyan. In panel (B) the pixels that are adjacent to the seeding pixel are added to the island as they are brighter than $\sigma_{\rm f} = 4$. Pixels adjacent to the island are now considered. The process is repeated in panel (C). In panel (D) some of the adjacent pixels are now below $\sigma_{\rm f}$ and are thus not added to the island, and are flagged as background pixels. In panel (E) there are no longer any pixels adjacent to the island which have not been rejected so the search for new pixels halts. In panel (E) there are no longer any pixels above the seeding limit of $\sigma_{\rm s} = 5$ so all remaining pixels are flagged as background pixels (panel F).

The operation of FloodFill is invariant to changes in the order in which the seed pixels are chosen. The output of FloodFill is a disjoint list of islands, each of which contains contiguous pixels that are above the $\sigma_{\rm f}$ limit. FloodFill does not perform any source characterization, although it is able to report the flux of an island of pixels by summing the pixel intensities. The fluxes that are reported by FloodFill have a positive bias which can be corrected as described by Hales et al. (in preparation).

3.6 AEGEAN

FloodFill forms the basis for two new source-finding algorithms: BLOBCAT (Hales et al., in preparation) and AEGEAN. Both algorithms begin with a set of islands identified by FloodFill but characterize these islands differently. The BLOBCAT program characterizes each island of pixels without assuming a particular source structure, where as AEGEAN assumes a compact source structure in order to fit multiple components to each island. Here we outline the AEGEAN algorithm, with a detailed description differed to Section 7.

The AEGEAN algorithm has been implemented in PYTHON and uses FloodFill to create a list of islands of pixels. AEGEAN was set to use a single background threshold of 5σ to seed the islands, and a flood threshold of 4σ to grow the islands. This threshold was set to $5\sigma = 125\,\mu{\rm Jy\,beam}^{-1}$ so that we are detecting sources above an SNR of 5.

AEGEAN uses a curvature map to decide how many Gaussian components should be fit to each island of pixels, and the initial parameters for each component. A PYTHON implementation of the MPFIT library[2] is used to fit the Gaussian components with appropriate constraints. Each island of pixels is thus characterized by at least one Gaussian component.

3.7 Source finding in radio surveys

The process of creating a catalogue of sources from survey images involves more than running one of the source-finding packages described in Section 3. Since the observing strategy, hardware and data reduction techniques can vary widely between surveys, standard source-finding packages are typically used only as a *starting point* for the creation of a source catalogue. The time required to carry out the observations for a large area sky survey is typically spread over multiple years. This prolonged observing schedule is usually accompanied by multiple iterations of calibration, data reduction and source detection, so that by the time the final observations are complete it is possible to produce a survey catalogue using a source-finding pipeline that has been refined over many years.

The NVSS (Condon et al. 1998), drew upon observations from 1993 to 1996, during which time the AIPS source-finding routine SAD was modified to create VSAD. The survey strategy for the NVSS was devised to give noise and sidelobe characteristics that were both low, and consistent across the sky. The configuration of the Very Large Array (VLA) was varied across the sky to ensure a consistent resolution throughout the survey. The survey strategy was thus designed to produce images that were nearly uniform across the sky, making the task of source finding as easy as possible. The completeness and reliability of the NVSS catalogue was improved in the years subsequent to the completion of the survey with the final stable release in 2002.

The SUMSS (Mauch et al. 2003) and Molonglo Galactic Plane Survey 2 (MGPS-2; Murphy et al. 2007) both used the source-finding package VSAD, however, the single purely east–west configuration of the telescope meant that the resolution varied with declination, and the regularly spaced feeds produced many image artefacts. The changing resolution and image artefacts meant that the source-finding algorithm produced many false detections. The

[2] http://code.google.com/p/agpy/source/browse/trunk/mpfit/?r=399

image artefacts appeared as radial spokes or arcs around bright sources. In order to rid the source catalogue of falsely detected sources, a machine learning algorithm was implemented (Mauch et al. 2003; Murphy et al. 2007). The machine learning algorithm was able to discriminate between real and false sources, but required substantial training to achieve high completeness and reliability.

An archival transients survey has recently been completed using the data from the SUMSS (Bannister et al. 2011). In an archival search spanning 20 yr of observations, the need for a fast source detection pipeline is not important, as fast transients will not be detected, and slow transients will remain visible for years to come. In the Bannister et al. (2011) survey, regions of sky with multiple observations were extracted from the archival SUMSS data. These regions of sky were analysed for sources which either changed significantly in flux, or which were detected in only a subset of the images. The SFIND package was used to detect sources, which were then remeasured using the MIRIAD routine IMFIT. A complication that was encountered in the analysis of the SUMSS data, was the contamination of candidate source lists due to source-finding errors. False positive detections and missed real sources both appear as sources which are only detected in a subset of all the images, and thus appear to be transient sources. The light curve of each transient event therefore needed to be double checked in order to remove such occurrences.

A similar transient detection project was carried out with new observations from the Allen Telescope Array (ATA), in the ATA Transients Survey (ATATS; Croft et al. 2011). Sidelobe contamination in the ATA images is much lower than that in the SUMSS images used in the Bannister et al. (2011) study, however, falsely detected sources in the individual images still resulted in false transient detections and required further processing to remove.

The process of finding sources and creating a catalogue extends beyond the operation of a source-finding package and has previously required substantial manual intervention. The next generation of telescopes, particularly the dedicated survey instruments, will be able to complete observations on a much shorter time-scale than current generation telescopes, and thus the time spent creating the refining the source catalogue will become a larger fraction of the total effective survey time. Source-finding packages that are able to produce more accurate, complete and reliable catalogues will provide a better starting point for the final version of the survey catalogue.

4 TEST DATA

We used a simulated data set to evaluate the source-finding algorithms described in Section 3. A simulated data set has the advantage that we are able to control the image properties (such as rms noise) and that we know the input catalogue.

Matching recovered sources with a true list of expected sources is an important part of the analysis presented in this paper. With any real data set, the list of expected sources comes with some degree of uncertainty, in that these lists are recovered from incomplete and noisy reconstructions of the radio sky. To avoid such uncertainties we generated a master catalogue of sources, which was then used to create a simulated image of the sky. With absolute control over the input catalogue and image characteristics, we are able to make more definitive statements about the quality of the catalogues that are produced.

The master source catalogue was generated with the following constraints.

(i) Fluxes: the source peak flux is distributed as $N(S) \propto S^{-2.3}$, and within the range (25 μJy, 10 Jy).

(ii) Positions: sources were randomly distributed in space within one of 10 regions of sky similar to that in Fig. 2. Source clustering was not considered.

(iii) Morphologies: the major and minor axes of each source were randomly distributed in the range 0–52 arcsec with position angles in the range ($-90^\circ, +90^\circ$).

A simulated sky image was created, which contained each of the sources in the input catalogue. The image has a 30-arcsec synthesized beam, and a 25 μJy beam^{-1} rms Gaussian background noise. The sources were injected with a peak flux and morphology as listed in the catalogue. The size of the image is 4801 × 4801 pixels with a scale of 6 arcsec pixel^{-1}, resulting in a synthesized beam sampling of 5 pixels beam^{-1}. Regions of sky exterior to the catalogue contain noise but no sources.

The simulated data set can be found online at www.physics.usyd.edu.au/hancock/simulations.

5 SOURCE-FINDING EVALUATION

The source-finding packages described in Section 3 were used to generate a catalogue of sources from the simulated images. Each source-finding package was run with a 5σ threshold. In the case of SFIND, the FDR was chosen so that the resulting threshold was equal to 5σ.

The source-finding algorithms were evaluated by comparing these catalogues with the input source catalogue. Three standard metrics that have been used in the comparison of catalogues, and hence source finders, are the completeness, reliability and flux distribution, as defined and discussed in Sections 5.2–5.5.

5.1 Cross-matching of catalogues

Much of the analysis that will be discussed in Sections 5.2–5.5 relies on the cross–identification of sources from two catalogues. A common criterion for accepting cross–identifications between catalogues is to choose the association with the smallest sky separation, up to a maximum matching radius. To decrease the chances of false associations we also consider the flux of the source when choosing between multiple matches within a matching radius of 30 arcsec. The distance in phase space, *D*, is given by

$$-\log(D) = \frac{(\alpha_1 - \alpha_2)^2}{\sigma_\alpha^2} + \frac{(\delta_1 - \delta_2)^2}{\sigma_\delta^2} + \frac{(S_1 - S_2)^2}{\sigma_S^2}, \qquad (1)$$

where (α, δ) are (RA, Dec.), *S* is the flux, and $\sigma_\alpha = \sigma_\delta = 30$ arcsec is the size of the convolving beam and $\sigma_S = 25$ μJy beam^{-1} is the image rms noise.

5.2 Flux distribution

The analysis of the flux distribution of a catalogue does not require catalogues to be cross-matched. Since the input source catalogue was constructed with a particular flux distribution, we should expect to see this distribution replicated in the output catalogues. Fig. 3 shows the flux distribution for each of the source finders compared to the input distribution. Except for SELAVY, each of the catalogues have a flux distribution that is consistent with the input catalogue. An excess of sources can be a sign of spurious detections, whilst a lack of sources can be due to incompleteness. If a source-finding algorithm has a flux distribution that deviates from the ideal case,

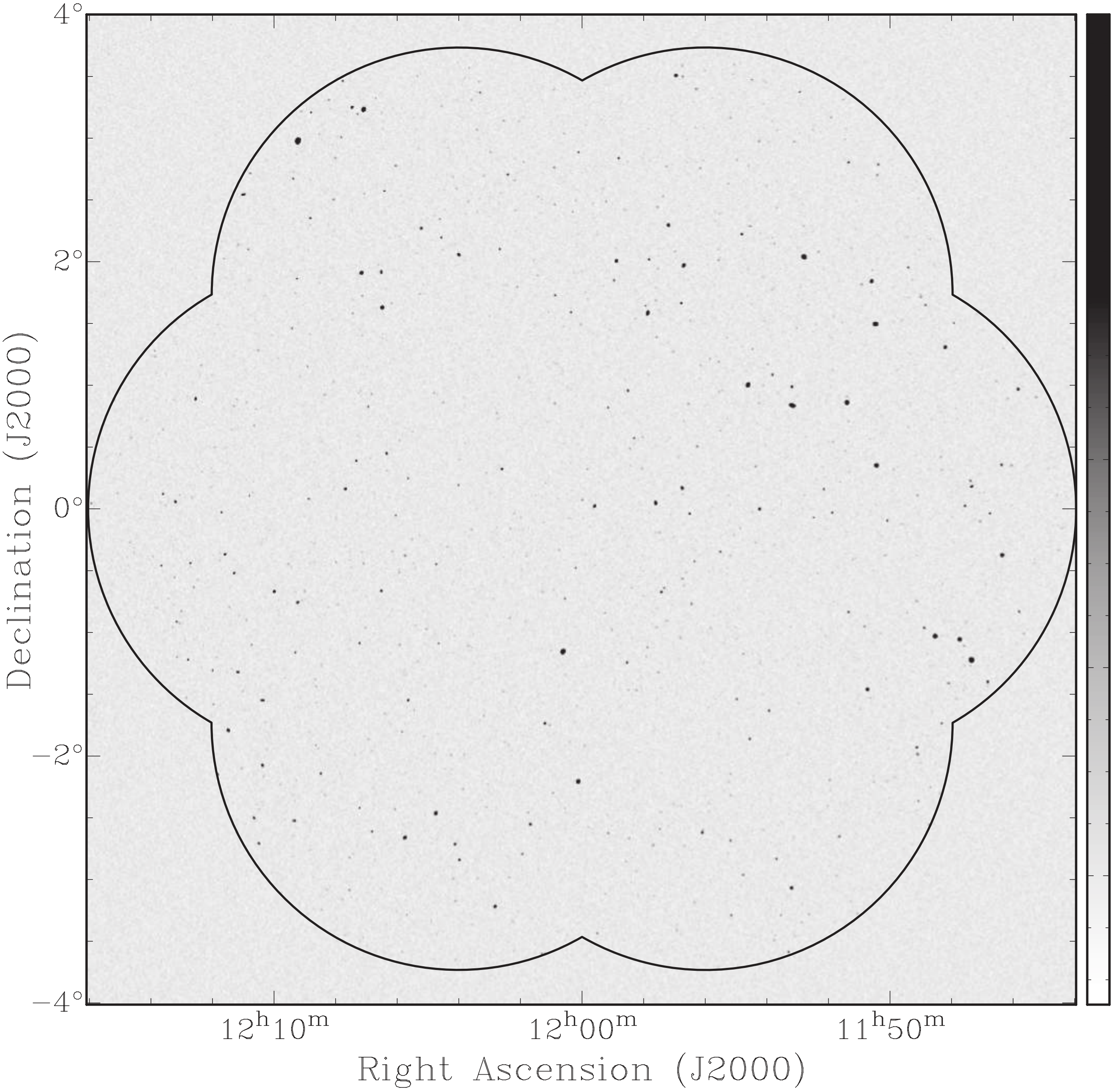


Figure 2. Simulated image of the sky. The black line delineates the region of sky containing the injected sources. The colour bar ranges from -5σ to 33σ.

it indicates that something is wrong, however, the cause of the problem cannot be identified from this graph alone. SELAVY has around double the number of sources at all flux levels as it suffers from source fragmentation. Since the fragmented components are close to the true position of the original source, the completeness and reliability of SELAVY are only partly compromised.

5.3 Completeness

The completeness of a catalogue at a flux S_0 is often defined as the fraction of real sources with true flux $S \geq S_0$ that are contained within the catalogue. In practice the completeness is measured as the number of sources with a *measured* flux $S \geq S_0$ that are contained within the catalogue. The two measures are comparable at large SNR, but when the SNR is ~5 the flux of a source can be in error by ~20 per cent. The completeness relative to the *measured* source fluxes is also effected by Eddington (1913) bias, whereas the completeness relative to the *true* source fluxes is not.

The completeness of a source finder was determined by matching the simulated catalogue with each of the source-finding catalogues. The completeness of a catalogue at a flux S_0 is then the fraction of real sources of flux greater than S_0 which are contained within the given catalogue. Fig. 4 shows completeness as a function of injected SNR for each of the source finders. Plotted alongside each of the completeness curves is a theoretical expectation of completeness for comparison. The expected completeness has been determined by taking each of the sources in the input catalogue and calculating the probability that it will be seen at a particular flux level, given the

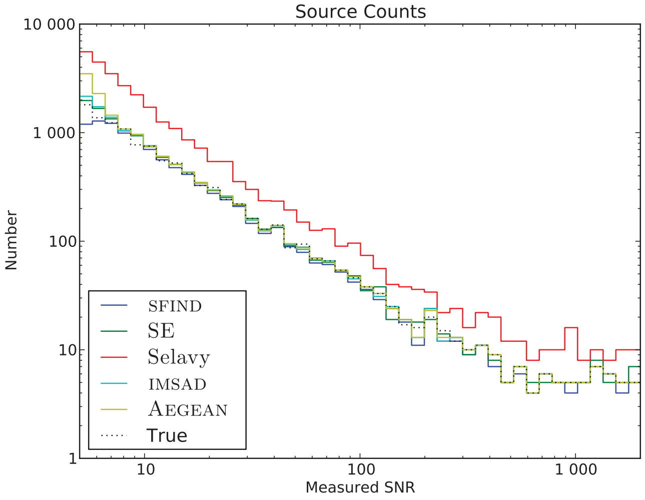


Figure 3. The source count distribution of the catalogued sources. The true source count distribution is shown as a dotted line. SELAVY consistently reports twice the true number of sources.

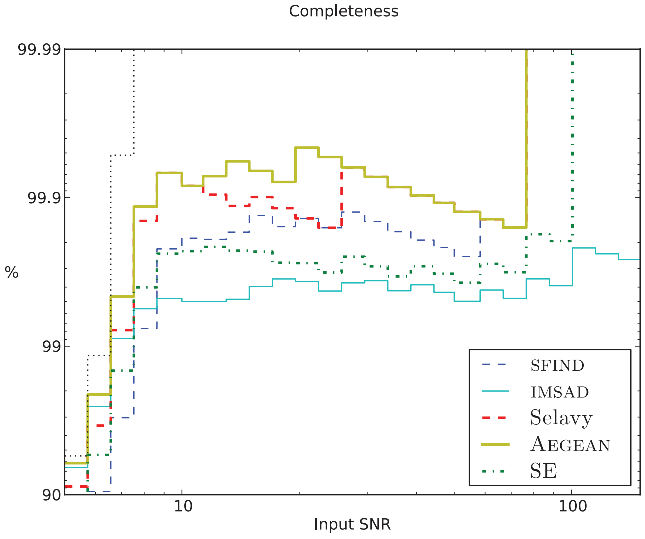


Figure 4. The completeness of each catalogue as compared to the input catalogue. The coloured curves represent the completeness of the named source finders. The black dotted curve represents the expected completeness of an ideal source finder as calculated by equations (2) and (3).

known rms in the image. The expected completeness is calculated as

$$\mathcal{C}(S_0) = \frac{\sum_S P(S > S_0)}{\sum_{S>S_0} N(S)}, \quad (2)$$

$$P(S > S_0) = \frac{1}{\sigma}\sqrt{\frac{4\log 2}{\pi}} \int_{S_0}^{\infty} \mathrm{e}^{-4\log 2\left((S'-S)/\sigma\right)^2} \mathrm{d}S'$$
$$= \frac{1}{2}\mathrm{Erfc}\left(\frac{\sqrt{4\log 2}(S_0 - S)}{\sigma}\right), \quad (3)$$

where $P(S > S_0)$ is the probability that a source of flux S will be seen at a flux greater than S_0 after noise has been included, and $N(S)\,\mathrm{d}S$ is the number of sources with flux between S and $S + \mathrm{d}S$. Erfc is the complementary error function. At SNR > 6 all of the source-finding packages produce catalogues that are greater than 99 per cent complete. The high completeness is due to, and responsible for, the wide-spread use of the given source finders. The different levels of completeness shown in Fig. 4 is a direct result of the source-finding algorithms implemented by each of the packages. The performance of SFIND is comparable to the other source finders above an SNR of 7, but less complete below this SNR. The lower completeness is a result of SFIND's focus on minimizing the FDR, as is shown in Fig. 5. SELAVY and AEGEAN are the most complete source-finding packages at all SNRs, however, SELAVY achieves this at a cost of an increased FDR (see Section 5.4 and Fig. 4). AEGEAN is able to achieve high completeness and low FDR at all SNRs. The completeness of each of the source-finding packages is summarized in Table 1.

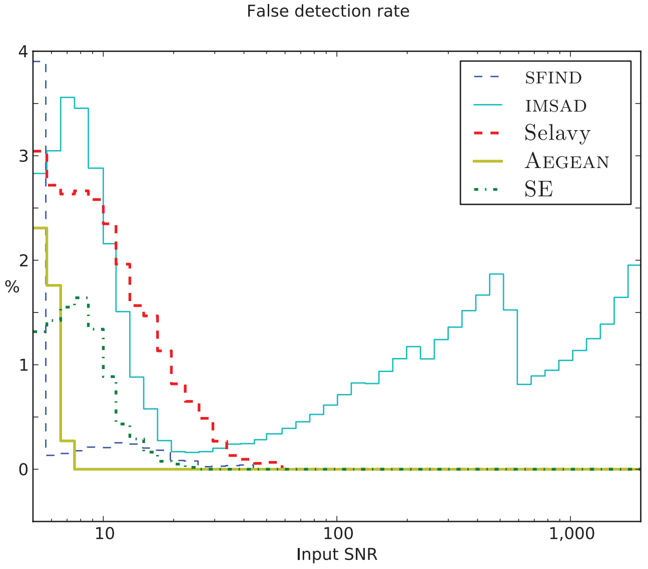


Figure 5. The FDR for each of the source-finding algorithms. The FDR is entirely a function of the source-finding algorithm. No falsely detected sources are expected above an SNR of 5 for the area of sky simulated.

Table 1. The completeness and reliability of each of the source-finding algorithms. The 'ideal' case has been included for comparison.

Package	Completeness (per cent)			Reliability (per cent)		
	5σ	10σ	50σ	5σ	10σ	50σ
IMSAD	93.44	99.50	99.49	97.17	97.75	99.66
SELAVY	91.20	99.92	99.87	96.96	97.65	99.93
SEXTRACTOR	88.31	99.77	99.62	98.68	99.11	100.0
SFIND	82.48	99.81	99.75	96.09	99.79	100.0
AEGEAN	93.87	99.91	99.87	98.69	100.0	100.0
Ideal	94.51	100.0	100.0	100.0	100.0	100.0

5.4 False detection rate

The FDR of a source-finding package at a flux S_0, is defined as the fraction of catalogued sources with $S \geq S_0$ which are not identified with a real source. The FDR of a source finder was determined by matching the resulting catalogue with the simulated source list. Catalogued sources which are not within 30 arcsec of a true sources are considered false detections. The FDR of a source-finding algorithm is related to the commonly used metric of reliability by

$$\mathrm{FDR} + \mathrm{Reliability} = 100\,\text{per cent}. \quad (4)$$

In Fig. 5 the FDR is plotted as a function of SNR for each of the source-finding packages. Substituting a flux of $S = 0\,\mu$Jy into equation (3) and considering the area of sky covered by the simulated images we expect that there is less than one false detection due to random chance. Thus an ideal source-finding algorithm should have an FDR of zero. The >1 per cent FDR peaks shown in Fig. 5 (especially for IMSAD) are due to islands of multiple sources that have not been properly characterized. SEXTRACTOR, IMSAD and SELAVY have a higher FDR than SFIND and AEGEAN, as the former are not able to accurately characterize islands of pixels.

The single sources that are fragmented into multiple sources by SELAVY often have positions that are close enough to the true position that they are not considered false detections, and thus do not significantly impact the FDR. However at low SNRs, SELAVY breaks single sources into three or even four components, and one or more of these components have a position distant enough from the true source that they are registered as false detections. This is evident in Fig. 5.

IMSAD suffers from the reverse problem to SELAVY, in that it will never break islands into multiple components even when they contain multiple sources. The position that is reported by IMSAD in such situations can be sufficiently far from the true position that these islands are registered as false detections. All of the false detections for IMSAD above an SNR of 20 in Fig. 5 are due to this flaw.

The reliability of each of the source-finding packages is summarized in Table 1.

5.5 Measured parameter correctness

For all measured catalogue sources that were identified with a true source it is possible to compare the measured parameters to the known true values.

Fig. 6 shows the median absolute deviation (MAD) in position, as a function of SNR, for each of the source-finding algorithms. The MAD is calculated for each SNR bin and is not a cumulative measure. An ideal source-finding algorithm will have a typical error in position that is proportional to C/SNR^2, where C is a constant that depends on the morphology of the source and the convolving beam (see Condon 1997 for detailed a calculation). The MAD in position of an ideal source-finding algorithm is calculated semi-analytically by assuming that each source in the input catalogue has measurement error of C/SNR^2. This ideal curve is plotted in Fig. 6.

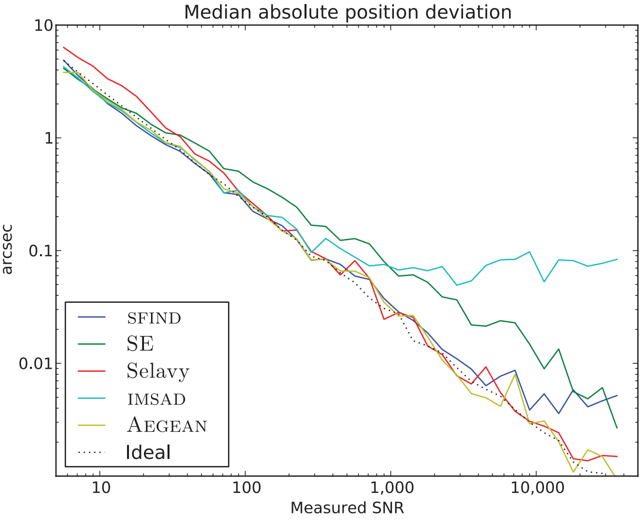


Figure 6. The accuracy with which each of the source-finding packages determines the position of sources. The dotted grey curve is the expected accuracy for an ideal elliptical Gaussian fit.

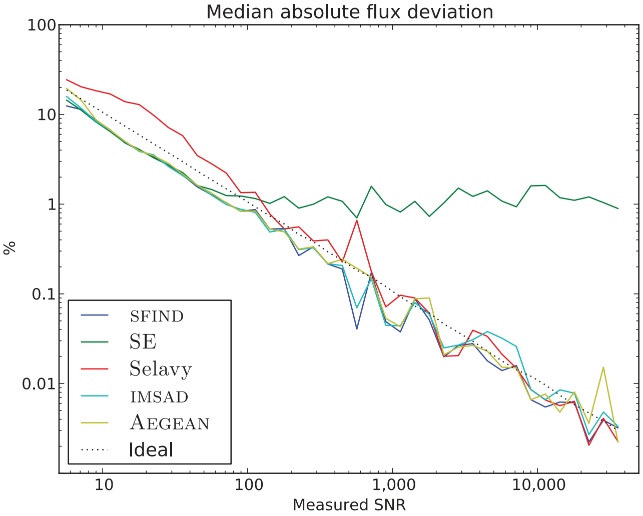


Figure 7. The accuracy with which each of the source-finding packages measures the flux of sources as a function of the reported flux. The dotted grey line is the expected accuracy for an ideal elliptical Gaussian fit.

As is expected, the accuracy with which a source position can be measured increases with flux, and is in agreement with the performance of an ideal Gaussian fitting routine, which is shown as a dotted curve in Fig. 6. The deviations from ideal behaviour that can be seen in Fig. 6 for the various source finders at high SNR are artefacts of the reporting accuracy of the packages. For example, IMSAD reports positions to a resolution of 0.1 arcsec and therefore cannot achieve a median absolute deviation in position better than ~0.1 arcsec. SFIND has similar problems at an SNR of $\gtrsim$3000. The median absolute position deviation for AEGEAN and SELAVY will also deviate from ideal, but at an SNR in excess of the 40 000 reported in Fig. 6. SEXTRACTOR does not use Gaussian fitting to characterize source positions and therefore does not perform as well as the ideal at SNR greater than 50. At an SNR of <100 SELAVY has a median absolute position deviation that is higher than the ideal. This is because of source fragmentation.

Fig. 7 shows the MAD in flux as a fraction of total flux, as a function of SNR. Again the ideal behaviour of a Gaussian fitter has been shown by a dashed curve. Overall the source-finding packages report fluxes that are consistent with the expected ideal Gaussian fit, the exceptions being SEXTRACTOR above an SNR of 50, and SELAVY at an SNR below 50. SEXTRACTOR deviates from the ideal and has a plateau at 1 per cent flux accuracy. In this work we use the corrected isophotal fluxes (FLUX_ISOCOR) from SEXTRACTOR. Of all the methods that are available for measuring fluxes in radio synthesis images, the corrected isophotal fluxes was found to be the most accurate. SELAVY deviates from the ideal case and has a flux accuracy of about 1/2 of ideal. The cause of this deviation is source fragmentation in which each component has only a fraction of the total true flux (see Section 5.2 and Fig. 3).

5.6 Initial evaluation summary

Each of the source-finding algorithms conforms to a high standard of completeness and reliability, and is able to produce a robust catalogue of a statistically large number of sources, with accurate measurements of position and flux. The completeness of the AEGEAN source-finding package is as good as or better than any of the other packages, and has been achieved without sacrificing reliability. In the context of next generation radio surveys, we are interested in the small differences between each of these source finders, and in how to optimize the approach to avoid even the residual small level of incompleteness and FDR. In surveys such as EMU (Norris et al. 2011), with an expected 70 million sources, an FDR of even 1 per cent translates into 700 000 false sources. This clearly has an impact on the study of rare or unusual behaviour. In particular we are interested in how the source-finding algorithm affects the final output catalogue at a level that is far more detailed than previously explored. With this in mind we now delve into specific cases in which existing source-finding packages fail.

6 MISSED SOURCES

We are now at the stage where we can consider the real sources that were missed by the source-finding packages, as well as the false detections that these programs generate. There are two populations of sources that are missed by one or more of the source-finding packages as will be discussed in Sections 6.1 and 6.2.

6.1 Isolated faint sources

In the simulated image, for which no clustering was taken into account, 99.5 per cent of the islands contained a single source.

The first population of sources that was not well detected by the source-finding algorithms are isolated faint sources. These sources have a true flux greater than the threshold, but have few or no pixels above the threshold due to the addition of noise. SFIND and SEXTRACTOR require an island to have more than some minimum number of pixels for it to be considered a candidate source. IMSAD, SELAVY and AEGEAN have no such requirement. The number of sources that are not seen in a catalogue due to the effects of noise can be calculated directly and is essentially the inverse problem to that of false detections. A correction can be applied to any statistical measure extracted from the catalogue in order to account for these missed sources. The only way to recover all sources with a true flux above a given limit is to have a threshold that is well below this limit, either by producing a more sensitive image or by accepting a larger number of false detections. Since this noise affected population of sources cannot be reduced by an improved source-finding algorithm, and can be accounted for in a statistically robust way, we will consider this population to be non-problematic.

6.2 Islands with multiple sources

The second population of sources that is not well detected by the source-finding packages are the sources that are within an island of pixels that contains multiple components. Examples of such islands are shown in Figs 8–10. If a source-finding algorithm is unable to correctly characterize multiple sources within an island, some or all of these sources will be missed. There are two approaches used by the tested algorithms to extract multiple sources from an island of pixels – iterative fitting and de-blending. Each of these approaches can fail to characterize an island of sources for different reasons, and will now be discussed in detail.

6.2.1 *Iterative fitting*

The first approach to characterizing an island of multiple components is an iterative one which relies on the notion of a fitting residual. The fitting residual is the difference between the data and the model fit. In the iterative approach a single Gaussian is fit to the island and the fitting residual is inspected. If the fitting residual meets some criterion then the fit is considered to be ‘good’ and a single source is reported. If the residual is ‘poor’ then the fit is redone with an extra component. Once either the fitting residual is found to be ‘good’ or some maximum number of components has been fit, the iteration stops and the extracted sources are reported. A disadvantage of this method is that if the number of allowed Gaussians (n) is poorly chosen, islands containing single faint sources can have a ‘better’ fitting residual when fit by multiple components, and source fragmentation occurs. When a source is fragmented it is difficult to extract the overall source parameters from the multiple Gaussians that were used in the fitting of the source. In particular the source flux is not simply the sum of the flux of the fragments. If the chosen value of n is too small then not all of the sources within an island will be characterized. These uncharacterized sources will contaminate the fitting of the previously identified sources resulting in a poor characterization of the island.

When the flux ratio of components within an island of pixels becomes very large, an iterative fitting approach can fail. The cause of this failure is related to the performance of an ideal Gaussian fitting routine. Fig. 7 shows the fractional error in measuring the amplitude of a Gaussian. For high SNR sources, the absolute flux error can be orders of magnitude below the rms image noise, so it may be expected that the maximum flux in the fitting residual should also be at or below the rms image noise. However, the main contribution to the flux seen in the fitting residual is not from

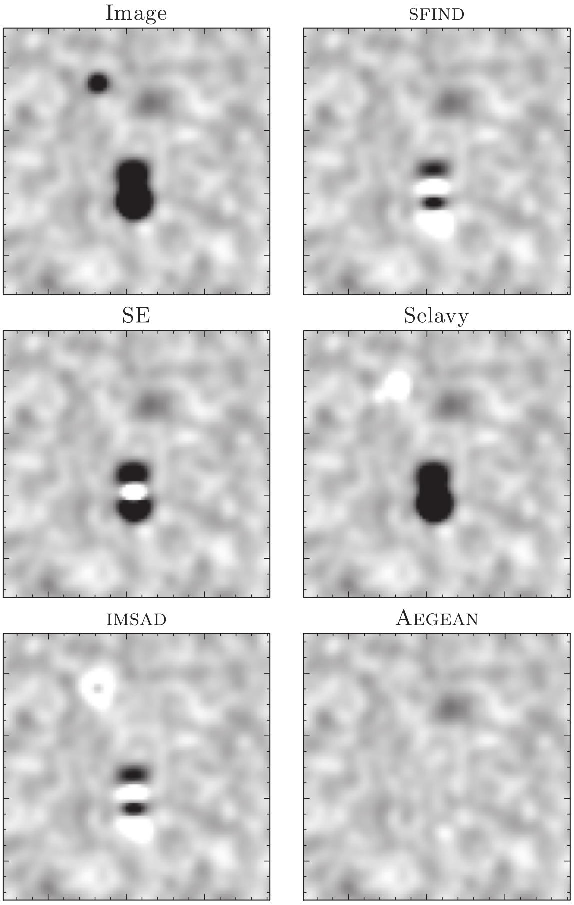


Figure 8. Top left: a section of the simulated image. Remainder: the fitting residual for each of the source-finding algorithms. AEGEAN was the only algorithm to fit all three sources, over both islands.

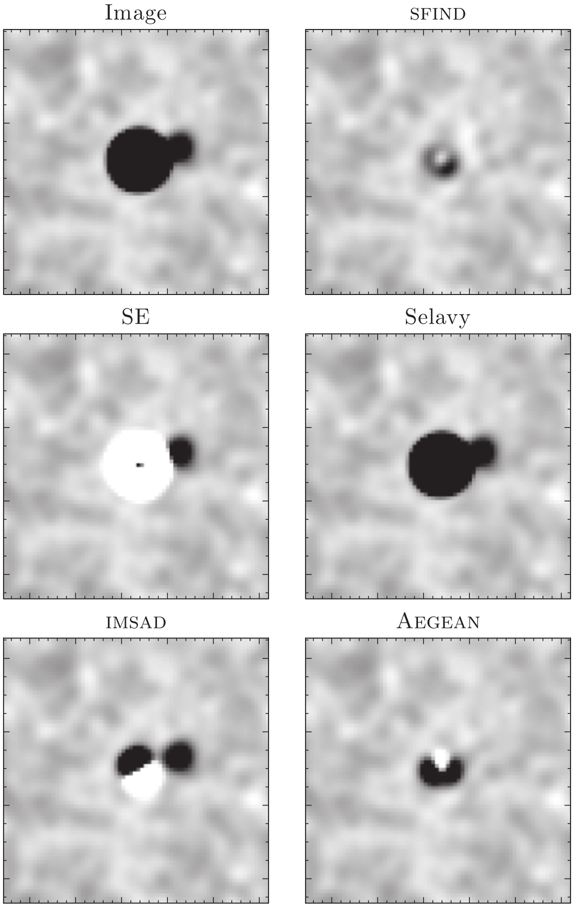


Figure 9. Top left: a section of the simulated image. Remainder: the fitting residual for each of the source-finding algorithms. AEGEAN and SFIND were able to correctly identify and characterize the two components but others were not.

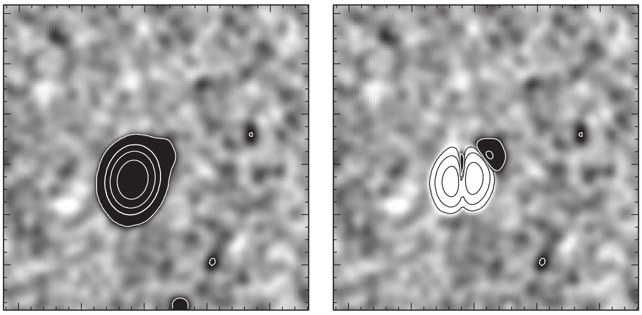


Figure 10. Left: an island of pixels from the simulated image containing both a 9 Jy source and a 1.7 mJy source. Right: the fitting residual formed by subtracting a (AEGEAN) fitted model of the 9 Jy source from the data. The pixel scale is -3σ (white) to $+5\sigma$ (black) with contours at an SNR of ±5, ± 50, ± 500 and ±5000 in contrasting tones. The flux of the source and its major axis have both been measured to within 0.05 per cent of the true value and yet the fitting residual has peaks at an SNR of over 500.

amplitude errors but from errors in estimating the full width at half-maximum (FWHM) of the source.

The amplitude difference between a (1D) Gaussian of amplitude A and FHWM of θ $(= 2\sqrt{2\log 2}\sigma)$ and a second Gaussian of identical amplitude A and FWHM of $\theta' = \theta + \Delta\theta$ is given by $F(x)$:

$$F(x) = A\left(e^{-(x^2 4\ln 2)/\theta^2} - e^{-(x^2 4\ln 2)/\theta'^2}\right), \tag{5}$$

which has maxima at

$$x_0^2 = \frac{\ln(\theta/\theta')}{2\ln 2}\left(\frac{\theta'^2\theta^2}{\theta'^2 - \theta^2}\right). \tag{6}$$

As a fraction of the true flux, the maximum residual is then

$$\frac{F(x_0)}{A} = \left(\frac{2\Delta\theta}{\theta}\right)\left(1 + \frac{2\Delta\theta}{\theta}\right)^{(\theta/2\Delta\theta)}. \tag{7}$$

The typical error in the measurement of θ is (Condon 1997)

$$\frac{\Delta\theta}{\theta} = \frac{\mu(\theta)}{\theta} \simeq \frac{\sigma}{A}, \tag{8}$$

so that a source with an SNR of A/σ will have a fitting residual with an SNR of

$$\frac{F(x_0)}{\sigma} = 2\left(1 + \frac{2\sigma}{A}\right)^{A/2\sigma}. \tag{9}$$

From equation (9) it is clear that in an island whose brightest source has an SNR of A/σ, sources below an SNR of $F(x_0)/\sigma$ will not be detected by an iterative fitting method. The fitting residual exceeds 5σ at an SNR as low as 11. Therefore, even an ideal Gaussian fitting routine will miss 5σ sources that are within the same island as a source of $\geq 11\sigma$ if an iterative approach is taken. If two Gaussian components are fit to an island of pixels such as that shown in Fig. 10, and the positions are left unconstrained, the fainter component will migrate towards one of the maxima in the fitting residual. The brighter source will then be characterized by two Gaussians, and the fainter source by none. The final result is that neither of the sources will be well characterized. It is therefore essential that a source-finding algorithm has some method for determining the number of Gaussian components within an island, as well as a way to stop the fitting process from mischaracterizing the two sources. A process called sectioning or de-blending is a common method.

6.2.2 *Sectioning or de-blending*

A second approach to characterizing islands with multiple sources is to use the distribution of flux within the island to determine the number of components to be fit, and then fit the components. This approach relies on some a priori knowledge of what a source looks like to break an island into components. SFIND, SEXTRACTOR and AEGEAN all use a form of sectioning to generate an initial estimate of the number of sources to be fit, as well as the starting parameters.

It is possible to create a statistical measure that will account for the number of sources that are missed because there are multiple sources within an island of pixels. This would, however, require detailed knowledge of the source-finding algorithm, the flux distribution of the source population, and the flux dependent two-point correlation function. The complexity of this calculation means that it is never computed and sometimes not even considered. Since many variable phenomena appear in or near known sources (e.g. radio supernovae in galaxies, extreme scattering events within our own Galaxy and more), an inability to accurately characterize this population of sources will make it difficult or impossible to reliably detect and characterize many variable events.

7 THE NEW SOURCE-FINDING PROGRAM: AEGEAN

With an understanding of how the underlying algorithms affect a source finder's ability to find and accurately characterize islands of pixels, we have created a new source-finding algorithm. The goal of the new algorithm is to incorporate the reliability and completeness performance of the packages studied in Sections 3–6, whilst improving on their ability to characterize islands of pixels. The source-finding algorithm is called AEGEAN, as it deals with many islands.

As background estimation and subtraction are not part of the focus of this work, AEGEAN has been designed with only a simple background estimation algorithm. For the analysis presented, AEGEAN was run with a detection threshold of 125 μJy beam^{-1}. AEGEAN uses the FLOODFILL algorithm described in Section 3.5 to create islands of pixels. The operation of Aegean is demonstrated in Fig. 11.

AEGEAN makes use of the notion of a single curvature map to characterize an island of pixels. The curvature κ of a function $f(x)$ is given by

$$\kappa = \frac{f''}{(1 + f'^2)^{3/2}} \tag{10}$$

(Reilly 1982). For a Gaussian with a FHWM of k pixels,

$$f'(x) = \frac{-16x \ln 2}{k^2} \mathrm{e}^{-(x^2 8 \ln 2)/k^2}, \tag{11}$$

so that f'^2 has a maxima at $x = k/\sqrt{2}$, and

$$f'^2 \leq \frac{1}{k^2} \frac{(\ln 2)^2}{2^9}. \tag{12}$$

For a Gaussian with $k \geq 1$, $f'^2 \ll 1$ and we can approximate

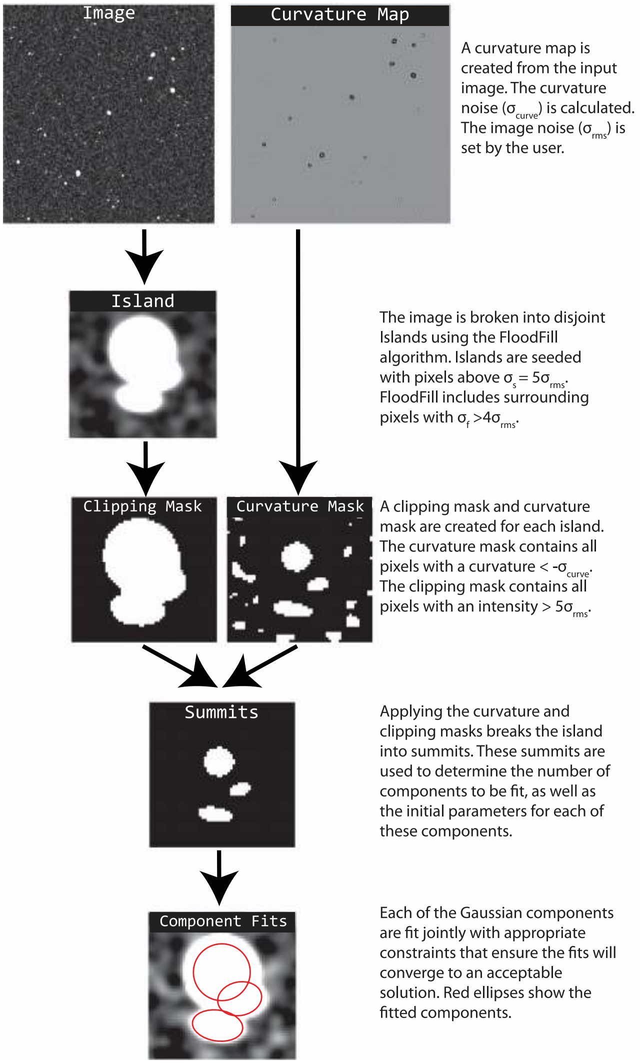
$$\kappa \simeq f''. \tag{13}$$

The curvature of a surface in a particular direction can be defined using equation (13), where the differentiation is along a unit vector in the chosen direction. Molinari et al. (2011) calculate the curvature of their input image in four image directions in order assist their source finding and characterization. We combine these four curvature measurements to calculate the mean curvature of an image. For an image convolved with a Gaussian with a FWHM of k pixels, the (mean) curvature, $\bar{\kappa}$ is equal to the mean of κ calculated in any two orthogonal directions (Reilly 1982). The discrete 2D Laplacian kernel,

$$L^2_{xy} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \tag{14}$$

calculates the sum of the second derivatives in four directions. Convolving the input image with L^2_{xy} will therefore produce a map of $2\bar{\kappa}$ – a single curvature map.

Islands of pixels are fit with multiple Gaussian components. The number of components to be fit is determined from a curvature map. The curvature map will be negative around local maxima. Groups of contiguous pixels that have negative curvature and fluxes above the threshold are called summits. An island of pixels will contain one or more summits. AEGEAN fits one component per summit, with the parameters of each of the components are taken from the corresponding summit. The position and flux are initially set to be equal to the brightest pixel within a summit, and the shape parameters (major/minor axis and position angle) are set to be the same as the convolving beam. Fig. 12 shows an example of two islands that contain multiple sources with the island boundaries and regions of negative curvature delimited. In the example in the left-hand panel of Fig. 12 there are three regions of negative curvature that are completely within the green island. This island is fit with three Gaussians. In the example in the right-hand panel of Fig. 12 there



A curvature map is created from the input image. The curvature noise (σ_{curve}) is calculated. The image noise (σ_{rms}) is set by the user.

The image is broken into disjoint Islands using the FloodFill algorithm. Islands are seeded with pixels above $\sigma_s = 5\sigma_{rms}$. FloodFill includes surrounding pixels with $\sigma_f > 4\sigma_{rms}$.

A clipping mask and curvature mask are created for each island. The curvature mask contains all pixels with a curvature $< -\sigma_{curve}$. The clipping mask contains all pixels with an intensity $> 5\sigma_{rms}$.

Applying the curvature and clipping masks breaks the island into summits. These summits are used to determine the number of components to be fit, as well as the initial parameters for each of these components.

Each of the Gaussian components are fit jointly with appropriate constraints that ensure the fits will converge to an acceptable solution. Red ellipses show the fitted components.

Figure 11. A demonstration of the operation of AEGEAN, using a single multiple component island as example. The input image is used to create a curvature map from which the curvature noise σ_{curve} is calculated. FLOODFILL is used to break the image into islands of pixels. The number of components in an island is estimated using a combination of the curvature mask and threshold mask. An elliptical Gaussian is fit to each of the components in the island simultaneously.

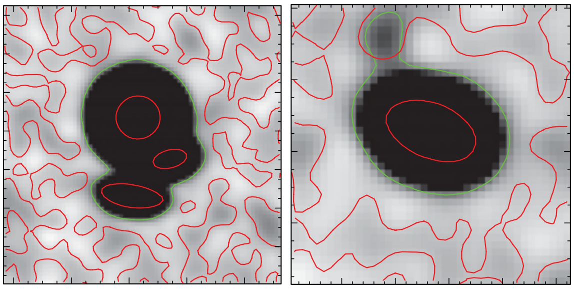


Figure 12. Two examples of the curvature analysis scheme. The grey-scale represents the flux density map and ranges from -3σ (white) to $+10\sigma$ (black). The green contour is at 5σ and represents the island boundary. The red contours are where the curvature map changes from positive to negative. Regions surrounded by a red contour have negative curvature and are the local maxima.

are two regions of negative curvature that overlap with the island of pixels. One component is contained entirely within the island, whilst the other is only partly within the island. Only the region of negative curvature that is within the green island is considered when estimating the initial parameters of the components. Both of the islands depicted in Fig. 12 contains a source that is bright enough that the expected fitting residual would be brighter than any of the other components within the island, and therefore an iterative fitting approach would only fit a single component (see Section 6.2.1). Since the island of pixels in the right-hand panel of Fig. 12 has two summits, aegean is able to accurately detect and characterize both components. Islands of pixels that contain only a single source have only a single summit and are fit with a single component.

To avoid faint components migrating to the fitting residual of brighter components, the position of each of the components is constrained to be within the corresponding summit. The flux of each component must be greater than 5σ. For low SNR sources, the true flux can be significantly different from the intensity of the brightest pixel in the summit, $S_{\max}$. For high SNR sources such noise variations are less important and beam sampling effects become more important.

For an image with a sampling rate of k pixels per beam a source of flux S which is located at the intersection of four pixels will effectively be sampled $\sqrt{2}(\theta/2k)$ pixels from the centre of the source. The intensity of the peak pixel is therefore given by

$$S_{\max} = S \exp\left(-\frac{\left(\sqrt{2}\theta/2k\right)^2 4\ln 2}{\theta^2}\right) \tag{15}$$

$$= S\,2^{-(2/k^2)}, \tag{16}$$

where θ is the FWHM of the source. The flux of each component is therefore constrained to be less than $S_{\max}\,2^{2/k^2} + 3\sigma$.

A Gaussian function has negative curvature from the peak out to $\pm\mathrm{FWHM}/\sqrt{2}$. The size of a summit is therefore used to constrain the component size. The major and minor axes of a component must be larger than the synthesized beam, and must remain smaller than $\sqrt{2}$ times the width of the summit. Beam sampling effects again play a role here, and so in aegean, we increase the limits on the major and minor axes each by two pixels to account for this. If the summit is smaller than the synthesized beam then the component is fit with the PSF.

The performance of aegean has been presented in Sections 5 and 6 along with the other source-finding algorithms under study.

8 CONCLUSIONS

Using a simulated data set, we have assessed the performance of some widely used source-finding packages, along with the ASKAPsoft source-finding program selavy. These source-finding packages are found to produce complete and reliable catalogues of isolated compact sources. We identify two populations of sources that are not well detected by the source-finding packages. The first population being faint sources close to the detection limit, and the second being sources which are within an island of pixels containing multiple components. Islands of pixels with multiple components are found to be poorly characterized by source-finding packages that take an iterative fitting approach to characterization. Source-finding packages that estimate the number of components in an island prior to fitting are less likely to mischaracterize the island. We have developed a new source-finding package, aegean, which is able to characterize the number of components within an island of pixels more accurately than any of the other packages tested.

aegean makes use of a curvature image which is derived from the input image with a Laplacian transform. Using the curvature image aegean is able to accurately determine the number of compact components within an island of pixels and produce a set of initial parameters and limits for a constrained fit of multiple elliptical Gaussians.

aegean has been shown to produce catalogues with a 5σ completeness that is better than our estimation of an ideal source finder. This completeness has been achieved without sacrificing reliability, and aegean is the most reliable of the tested algorithms. The next generation of radio surveys will be sensitive enough that ~5 per cent of the islands in the image will contain multiple components and therefore the ability to characterize such islands is of critical importance. aegean is able to accurately characterize islands of pixels which contain multiple compact components.

We have shown that in order to improve the reliability and completeness of source catalogues it is necessary to perform constrained multiple Gaussian fitting. An accurate estimation of initial parameters and sensible constraints are both critical when multiple component Gaussian fitting is performed. We have demonstrated a method for estimating and constraining the fitting parameters which is based on the curvature of the image. We anticipate that by adopting the aegean algorithm, the next generation of radio continuum surveys will be able to achieve more complete, reliable and accurate catalogues without relying on significant manual intervention.

ACKNOWLEDGMENTS

The authors thank David McConnell, David Kaplan and Theodora Papadimatos for useful discussions about source-finding algorithms. TM, BMG and PJH acknowledge support from the Australian Research Council through Super Science Fellowship grant FS100100033 and JRC through Discovery Projects grant DP1097291. The Centre for All–Sky Astrophysics is an Australian Research Council Centre of Excellence, funded by grant CE11E0090.

REFERENCES

Adams T. J., Bunton J. D., Kesteven M. J., 2004, Exp. Astron., 17, 279

Bannister K. W., Murphy T., Gaensler B. M., Hunstead R. W., Chatterjee S., 2011, MNRAS, 412, 634

Becker R. H., White R. L., Helfand D. J., 1995, ApJ, 450, 559

Bertin E., Arnouts S., 1996, A&AS, 117, 393

Chatterjee S., Murphy T., VAST Collaboration, 2010, BAAS, 42, 515

Condon J. J., 1997, PASP, 109, 166

Condon J. J., Cotton W. D., Greisen E. W., Yin Q. F., Perley R. A., Taylor G. B., Broderick J. J., 1998, AJ, 115, 1693

Croft S., Bower G. C., Keating G., Law C., Whysong D., Williams P. K. G., Wright M., 2011, ApJ, 731, 34

Eddington A. S., 1913, MNRAS, 73, 346

Hopkins A. M., Miller C. J., Connolly A. J., Genovese C., Nichol R. C., Wasserman L., 2002, AJ, 123, 1086

Huynh M. T., Hopkins A. M., Norris R. P., Hancock P. J., Murphy T., Jurek R., 2011, Publ. Astron. Soc. Australia, in press

Johnston S. et al., 2008, Exp. Astron., 22, 151

Lonsdale C. J. et al., 2009, Proc. IEEE, 97, 1497

Mauch T., Murphy T., Buttery H. J., Curran J., Hunstead R. W., Piestrzynski B., Robertson J. G., Sadler E. M., 2003, MNRAS, 342, 1117

Molinari S., Schisano E., Faustini F., Pestalozzi M., Di Giorgio A. M., Liu S., 2011, A&A, 530, A133
Murphy T., Mauch T., Green A., Hunstead R. W., Piestrzynska B., Kels A. P., Sztajer P., 2007, MNRAS, 382, 382
Norris R. P. et al., 2011, Publ. Astron. Soc. Australia, 28, 215
Reilly R. C., 1982, American Math. Mon., 89, 180
Roerdink J., Meijster A., 2001, Fundamenta Inf., 41, 187
Sault R. J., Teuben P. J., Wright M. C. H., 1995, in Shaw R., Payne H., Hayes J., eds, ASP Conf. Ser. Vol. 77, Astronomical Data Analysis Software and Systems IV. Astron. Soc. Pac., San Francisco, p. 433
Whiting M., 2012, MNRAS, in press

This paper has been typeset from a TEX/LATEX file prepared by the author.