Servicio Meteorológico Nacional
Argentina

Sistema de Control de Calidad de Datos de Radar en el Servicio Meteorológico Nacional - Parte II: Implementación Operativa

Nota Técnica SMN 2021-87

Maximiliano Sacco[1], Aldana Arruti[1], Paula Maldonado[1], Martin Rugna[1], Juan Ruiz[2], y Luciano Vidal[1]

[1] Dirección de Productos de Modelación Ambiental y de Sensores Remotos, Dirección Nacional de Ciencia e Innovación en Productos y Servicios.

[2] Centro de Investigaciones del Mar y la Atmósfera, CONICET, FCEyN-UBA.

Marzo 2021

Ministerio de Defensa
Presidencia de la Nación

Servicio Meteorológico Nacional
Argentina

Información sobre Copyright

Este reporte ha sido producido por empleados del Servicio Meteorológico Nacional con el fin de documentar sus actividades de investigación y desarrollo. El presente trabajo ha tenido cierto nivel de revisión por otros miembros de la institución, pero ninguno de los resultados o juicios expresados aquí presuponen un aval implícito o explícito del Servicio Meteorológico Nacional.

La información aquí presentada puede ser reproducida a condición que la fuente sea adecuadamente citada.

Servicio Meteorológico Nacional Argentina

Resumen

En la presente Nota Técnica se describe el esquema de implementación del sistema de control de calidad de datos de radar meteorológico descrito en Arruti y otros (2021). Se listan los requerimientos relevados de distintos usuarios para realizar una ejecución operativa, a continuación se describe el diseño de arquitectura elegida para responder las demandas del usuario. Finalmente, se detallan características de la implementación realizada y se evalúan los requerimientos necesarios de hardware simulando una situación posible del cotidiano operativo.

Abstract

This Technical Note describes the implementation scheme of the weather radar data quality control system described in Arruti et al. (2021). It lists the requirements gathered from different users to perform an operational implementation, and then describes the architecture design chosen to meet the user's demands. Finally, the characteristics of the implementation are detailed and the necessary hardware requirements are evaluated by simulating a possible situation of daily operation.

Palabras clave: radar meteorológico, control de calidad, reflectividad, implementación operativa

Citar como:

Sacco, M., A. Arruti, P. Maldonado, M. Rugna, J. Ruiz, L. Vidal, 2021: Sistema de control de calidad de datos de radar en el Servicio Meteorológico Nacional - Parte II: Implementación Operativa. Nota Técnica SMN 2021-87.

Servicio Meteorológico Nacional Argentina

1. INTRODUCCIÓN

La necesidad e importancia de realizar un control de calidad sobre los datos de radar meteorológico se analiza en detalle en Arruti y otros (2021) donde se utiliza un sistema integral que combina diferentes variables para producir, en una única salida, un conjunto de archivos donde se discrimina entre los ecos meteorológicos y los ecos no meteorológicos y las regiones sin datos válidos. En la presente Nota Técnica se detallan los desafíos y decisiones de diseño que fueron evaluadas a la hora de realizar la implementación operativa del sistema de control de calidad en el Servicio Meteorológico Nacional (SMN).

El código completo y funcional se puede encontrar en el repositorio Gitlab del SMN como QC-RadarV1.0 (https://gitlab.smn.gov.ar/ID/qc-radarv1.0). Esta implementación también fue diseñada para integrarse con la implementación del sistema de asimilación de datos detallada en Dillon y otros (2020).

2. RELEVAMIENTO Y ANÁLISIS DE REQUERIMIENTOS

Se realizaron varias entrevistas con los usuarios, técnicos e interesados en implementar una solución operativa ajustada a las posibilidades y necesidades del SMN. Como resultado de dichas entrevistas se identificaron un conjunto de requerimientos de usuarios, funcionales y no funcionales, que serán mencionados y analizados a continuación junto con un glosario de definiciones usadas para este contexto.

2.1 Glosario

Control de calidad: mecanismos y/o técnicas orientadas a mejorar la calidad de un dato o producto.

Técnica de filtrado: técnica o algoritmo parametrizable orientado a identificar, corregir, minimizar o eliminar ruido o información no deseada dentro de un conjunto de datos.

Filtro: es una técnica de filtrado y un conjunto de parámetros asociados. Es decir, una misma técnica de filtrado con dos conjuntos de parámetros distintos define dos filtros distintos.

Secuencia de filtros: define un encadenamiento ordenado específico de filtros que serán aplicados a los datos uno a continuación de otro. Esta definición asume que la ejecución en secuencia de un conjunto de filtros puede dar distintos resultados si el orden de la secuencia es alterado.

Dato de entrada: hace referencia al archivo de datos de radar al que se desea aplicar el control de calidad.

Producto: es el resultado obtenido luego de aplicar el control de calidad a un conjunto de datos de entrada. Específicamente para esta Nota Técnica, nos referimos a los archivos resultantes de aplicar una secuencia de filtros a un archivo de datos de radar de entrada.

Operador: responsable de controlar el correcto funcionamiento del sistema y de configurarlo según la demanda de los usuarios.

Usuario: Responsable de definir los productos y los niveles de calidad requeridos para los mismos.

Servicio Meteorológico Nacional Argentina

2.2 Requerimientos funcionales

1) *El sistema debe contemplar un mecanismo que permita incorporar de manera sencilla nuevas técnicas de filtrado.*

2) *El sistema debe permitir, sin intervención del desarrollador, configurar secuencias de filtros sobre filtro existentes.*

3) *El sistema debe permitir de manera eficiente aplicar múltiples secuencias de filtros a un único dato de entrada generando así múltiples productos.*

   Al existir distintos filtros a los cuales se les pueden modificar diversos parámetros de configuración, existe la posibilidad que distintos usuarios requieran distintas parametrizaciones para la misma secuencia de filtros, distinta cantidad de filtros o distinto orden para la aplicación de los mismos. Además, al existir filtros que demoran más tiempo en ejecutarse, las exigencias de los usuarios debido a la demanda temporal puede requerir que no se ejecuten.

2.3 Requerimientos no funcionales

1) *Contemplar fallas en los datos de entrada*

   Los usuarios advirtieron que en ocasiones, los archivos de entrada llegan a los servidores del SMN con errores en el formato. En particular, algunos fabricantes de radares pueden enviar las variables en diferentes archivos (datos de entrada segregados en múltiples archivos) y en diferentes ventanas temporales o incluso nunca recibir el conjunto de archivos necesario para formar el conjunto completo de datos de entrada. Debido a estas condiciones externas, se identificaron dos escenarios frecuentes:

   - *Datos de entrada mal formados*. El sistema deberá detectar esta incidencia y descartar todo el conjunto de datos de entrada asociado.

   - *Contemplar retrasos en la llegada de archivos/variables.* El sistema debe realizar un seguimiento de las variables disponibles para cada volumen y ejecutar el control de calidad con la mayor cantidad de variables posibles (dado que esto resultará en un mejor campo de reflectividad corregida) en tiempo real. En el caso de que una variable llegue con retraso y que la configuración de los filtros lo requiera, el sistema debe poder reprocesar todo el volumen utilizando, nuevamente, la mayor cantidad de variables disponibles.

2) *Diseño e implementación escalable en funcionalidad.* Contempla la posibilidad de realizar el control de calidad sobre múltiples volúmenes de radar al mismo tiempo y con diferentes configuraciones de filtros. Se espera que en el corto y mediano plazo se agreguen o cambien requerimientos y demandas. Por esto, el sistema debe ser lo suficientemente flexible como para poder incorporar fácilmente situaciones como agregar más radares o nuevos filtros, modificar las demandas de los usuarios, entre otros.

Servicio Meteorológico Nacional Argentina

3) *Diseño e implementación escalable en software y hardware de base.* Debe contemplarse la posibilidad de ejecutar en múltiples configuraciones de hardware con mínima o ninguna intervención de los desarrolladores. Por ejemplo, el sistema tiene que ser capaz de correr en un servidor dedicado o en un cluster tipo beowolf (múltiples computadoras dedicadas a procesar un único problema) permitiendo de esta manera incrementar en poder de cómputo y en consecuencia la velocidad de procesado.

4) *Adaptar el código existente con el mínimo impacto.* Como se muestra en Dillon y otros (2020), un sistema de control de calidad de datos de radar se utiliza para procesar y generar observaciones útiles para el proceso de asimilación de datos. Un requisito para este sistema es incorporar los esfuerzos realizados de manera de reutilizar la mayor cantidad de código existente.

3. DISEÑO DE ARQUITECTURA

Con el fin de poder cumplir con el requisito de escalabilidad del software, se optó por utilizar una arquitectura Maestro/Trabajador (Master/Slave) ya que este diseño refuerza la división de tareas entre la gestión de los datos y el posterior procesamiento de la información. Además permite adaptar fácilmente el código a distintos escenarios de ejecución (servidor dedicado o cluster de computadoras) como se solicita en el requerimiento de escalabilidad de hardware.

La entidad *Master* periódicamente verifica si existe nueva información de radar para ser procesada y, en función de la configuración especificada, elabora un plan de tareas que será ejecutado por una o varias entidades *Worker*. Estas se podrán ejecutar en el mismo servidor donde se ejecuta *Master* o en otros nodos de cómputo designados para esta tarea. De esta manera se resuelve el requerimiento de escalabilidad permitiendo que la necesidad de más poder de cómputo (dado por nuevos filtros, más radares o más usuarios) se pueda resolver fácilmente incorporando nuevos nodos de procesamiento al sistema donde se puedan ejecutar los *Workers*. En la Figura 1 se muestra un esquema de la arquitectura implementada.

La planificación del *Master* se especifica mediante un conjunto de archivos de configuración específicos por productos, donde se determina los radares donde se quiere aplicar el control de calidad, la secuencia de filtros a considerar y las variables indispensables para iniciar el proceso. Estos archivos de configuración son visibles tanto por el *Master* como por los *Workers*.

4. DISEÑO DE IMPLEMENTACIÓN

La implementación de las entidades *Master* y *Workers* mencionadas en la arquitectura se desarrollaron en tres módulos, el módulo **Master**, el módulo **Worker** y el módulo **Despachar** que es el encargado de ejecutar los workers. Todos los módulos fueron desarrollados en código Python para la versión 3.6 o superior. A continuación se listan las librerías auxiliares de Python que se utilizan en la implementación:

Servicio Meteorológico Nacional Argentina

- os
- subprocess
- pandas
- numpy
- datetime
- configparser
- xml
- multiprocessing
- slurmpy
- pickle
- sys

4.1 Modulo Master.py

La funcionalidad de este módulo está descrita en la Figura 1 y se corresponde a la entidad *Master* en color verde donde cada recuadro identifica una funcionalidad implementada. Este módulo fue diseñado para ser activado en forma periódica (al momento de generar esta nota técnica, la activación del módulo es únicamente mediante la ejecución del mismo). En cada activación leerá una tabla (*dataframe* de Pandas) donde se resume el estado del sistema de la corrida anterior.

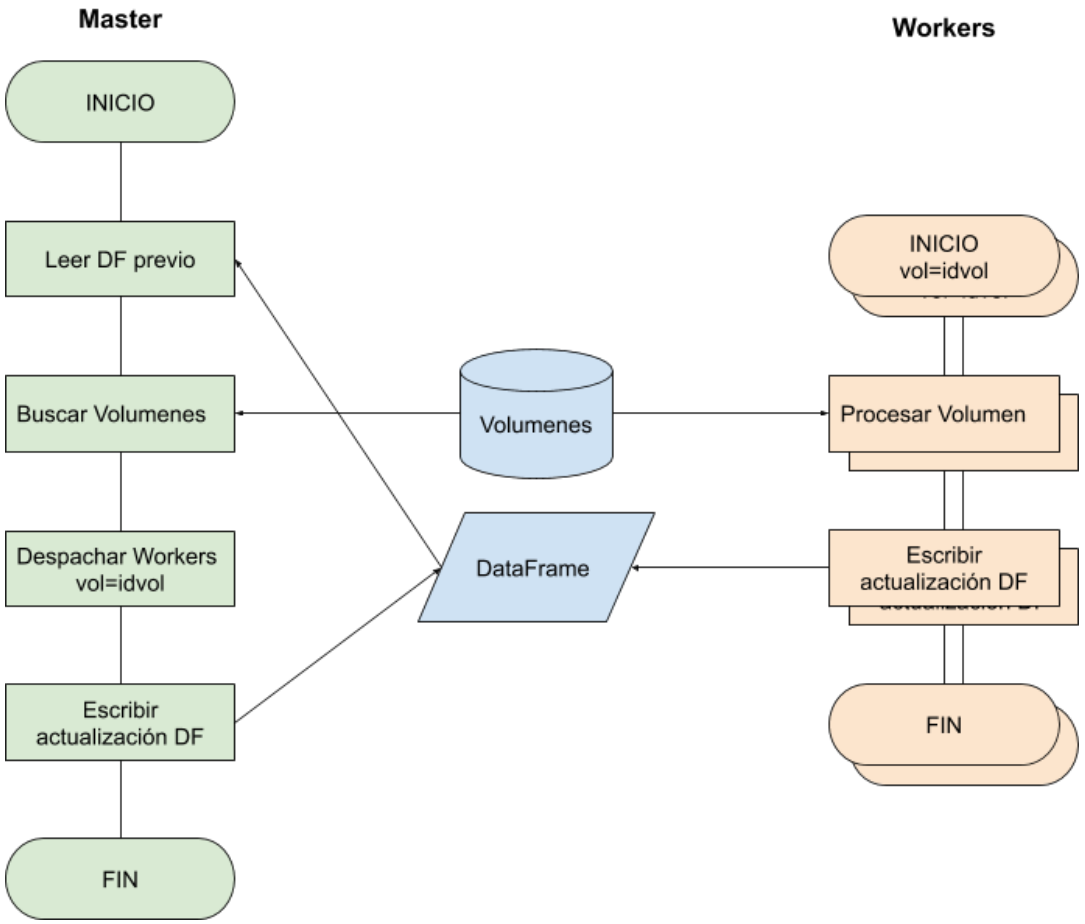

Fig. 1 *En el diagrama se muestra, en la columna izquierda el funcionamiento de la entidad Master, en la columna derecha el funcionamiento de las entidades Workers. En el centro se encuentran los elementos comunes a ambas entidades, donde Volúmenes representa el conjunto de datos de entrada y DataFrame es la tabla de tareas generadas por el Master para ser ejecutadas por los workers.*

Servicio Meteorológico Nacional Argentina

a) Función *Leer Dataframe*

El *dataframe* es una tabla que contiene información relevante para realizar el procesamiento; como el nombre del radar, la fecha de inicio del escaneo, la variable que contiene el archivo, el formato del archivo, entre otros elementos detallados en la Tabla 1.

Tabla 1: *Elementos utilizados en la tabla leída por el proceso Master.*

Variable	Descripción	Ejemplo
fecha_ini	Fecha y hora del inicio del escaneo en UTC	`"2021-02-01 20:46:43"`
file_input	Ruta completa al archivo crudo de radar	`"/ruta/al/archivo/RMA1_0301_01_DBZH_20210201T204643Z.H5"`
file_ext	Extensión del archivo crudo de entrada	`"H5"`
names	Nombre del archivo crudo de entrada	`"RMA1_0301_01_DBZH_20210201T204643Z.H5"`
radar_id	Identificador único del radar	`"RMA1"`
estrategia	Identificador de estrategia de escaneo volumétrico	`"0301"`
volumen	Identificador de volumen dentro de la estrategia	`"01"`
range	Rango de alcance máximo de los datos dada una estrategia y un volumen	`"240"`
variable	Variable del archivo crudo de entrada	`"reflectivity"`
config	Configuración de los filtros que se aplican y la parametrización de los mismos.	`"asimilacion"`
state	Estado de procesamiento del archivo	`"D"`
file_output	Nombre del archivo de salida del control de calidad	`"/ruta/al/archivo/cfrad.20210201_204643.RMA1_0301_01.nc"`

Cada registro de la tabla es una tarea que debe ser llevada a cabo por un *Worker*. Entonces, existe para una misma información de radar tantos registros como ***productos*** se hayan definido (configuraciones de usuarios, parametrizaciones de filtros, etc).

La tabla se guarda en un archivo en formato CSV (campos separados por comas) llamado ***df_qc-radar.csv*** que se encuentra en el directorio indicado por la variable [System ➡ BASEDIR] configurada en el archivo de

Servicio Meteorológico Nacional Argentina

configuración *"config.ini".* Ver sección "archivos de configuración" más adelante. La decisión de usar archivos de formato CSV es que estos pueden ser fácilmente interpretados por humanos y facilita el control y monitoreo de las actividades del *Master* manualmente.

b) Función *Buscar volúmenes*

Este procedimiento busca nuevos datos de entrada en el directorio indicado por la variable [System➡DATAROOT] en el archivo de configuración "config.ini" y genera un nuevo registro en la tabla de tareas.

Además, para cada archivo de producto, se leen las variables [BASE➡VARLIST], que indica toda la información que debe estar presente en los archivos de datos para generar un producto completo, y la variable [BASE➡VARLISTMIN], que indica la mínima información que se necesita para generar un producto con la calidad mínima aceptable, pero que de haber mayor información, el producto podría ser de mejor calidad.

Existen al menos cuatro escenarios posibles que pueden ocurrir al analizar los datos de entrada según la configuración de producto configurada.

1. **Datos de entrada completos para generar un producto**

   Para cada dato de entrada nuevo (que no figure en el tabla actual) y para cada archivo de configuración de producto (ver sección "configuración de producto"), si los datos disponibles alcanzan para genera un producto, entonces se genera un nuevo registro y se agrega a la tabla actual con el estado "D" de disponible.

2. **Datos de entrada suficientes para generar un producto**

   Para los productos que se necesiten generar lo más rápido posible, aun si la calidad final no es óptima, existe la posibilidad de generar un primer filtrado en caso de existir una cantidad de información mínima (VARLISTMIN). En este caso se genera una entrada en la tabla de Tareas.

3. **Actualización de datos de entrada**

   Este escenario es consecuencia inmediata del escenario 2, donde se generó una primera versión del producto con mínima información y a medida que el resto de la información (VARLIST) esté disponible, generar nuevas versiones del producto. En este caso se genera una entrada en la tabla de Tareas.

4. **Datos de entrada insuficientes para generar un producto**

   En este caso, no se generará un nuevo registro en la tabla de tareas hasta que la información disponible alcance para generar un producto (escenarios 1 y 2)

Al momento de generar esta Nota Técnica, se dispone de funciones para leer datos de entrada de los siguientes radares de la red operativa nacional (SINARAME e INTA; de Elia y otros 2017):

Servicio Meteorológico Nacional Argentina

- INTA Anguil (La Pampa)
- INTA Paraná (Entre Ríos)
- RMA1 - Córdoba (Córdoba)
- RMA2 - Ezeiza (Buenos Aires)
- RMA3 - Las Lomitas (Formosa)
- RMA4 - Resistencia (Chaco)
- RMA5 - Bernardo de Irigoyen (Misiones)
- RMA6 - Mar del Plata (Buenos Aires)
- RMA7 - Neuquén (Neuquén)
- RMA8 - Mercedes (Corrientes)
- RMA10 - Bahía Blanca (Buenos Aires)
- RMA11 - Termas de Río Hondo (Santiago del Estero)

.

c) Función *Despachar Workers*

Por requerimiento se solicitó al menos disponer de dos métodos para procesar los datos, uno que se ejecuta en un servidor exclusivo y aprovecha el paralelismo con memoria compartida y otro que se ejecuta en un cluster aprovechando el paralelismo con memoria distribuida.

Para cumplir con dicho requisito se diseñó el módulo *despachar.py*. Este módulo implementa una interfaz para independizar el proceso de gestión (generación de tabla de tareas) respecto del método que será utilizado para efectivamente realizar la tarea (ejecutar un *worker*).

Al momento de generar esta Nota Técnica, los dos métodos requeridos fueron implementados y se puede seleccionar uno o el otro mediante el archivo de configuración “config.ini”.

d) Función *Escribir DataFrame*

Finalmente, luego de despachar a todos los *Workers*, el *Master* revisa la lista de archivos y elimina archivos viejos (donde la antigüedad está dada por un archivo de configuración general). Finalmente, se guarda la tabla de tareas en el archivo ***df_qc-radar.csv*** para ser utilizado en una próxima ejecución del *Master*.

4.2 Modulo Worker.py

Cada uno de los *Workers* lanzados se ocupa de obtener los archivos de un determinado radar, para una fecha de inicio del escaneo y para una configuración (parámetros de inicialización del *Worker*). En función de esta última, procesa los filtros determinados allí con los parámetros elegidos y genera un único archivo de salida en formato NetCDF.

5. ARCHIVOS DE CONFIGURACIÓN

Existen dos clases de archivos de configuración. El archivo de configuración general del sistema ***config.ini,*** que es único para toda la instalación y los archivos de configuración de productos, que existe uno por cada producto que se desee generar.

Servicio Meteorológico Nacional Argentina

5.1 Configuración General - config.ini

El archivo ***config.ini*** está separado por secciones y a continuación describiremos las más importante.

1) **[System]**
   - **BASEDIR**: directorio base de instalación
   - **DATAROOT**: directorio base donde se encuentran los volúmenes de radar
   - **USERCONF**: lista de productos (nombres que estarán asociados al correspondiente archivo de configuración)
   - **THREADS**: threads disponibles para cada Worker
   - **MAXRADAR**: cantidad de Workers que serán ejecutados en simultáneo
   - **WINDOWS**: ventana de tiempo en minutos para búsqueda de datos
   - **MODO**: modo de ejecución de los workers (local | slurm)
   - **TOPO_RAW_DATAPATH**: ruta de los datos de topografía (formato tif)
   - **TOPO_RADAR_DATAPATH**: ruta de los datos de topografía interpolada
   - **TOPO_FILE_EXT**: extensión de los archivos de topografía interpolada
   - **UNDEF**: valor de undef que tendrán los pixeles eliminados por los filtros
   - **CONDABIN**: ruta (path) de conda a utilizar
   - **CONDAENV**: entorno de conda a utilizar
2) **[SLURM]**
   - **PARTITION**: nombre de la cola usada para procesar
   - **MAXTIME**: tiempo máximo permitido para procesar en los nodos
   - **MAXNODE**: máxima cantidad de nodos disponibles para la ejecución
   - **CORES**: cantidad de cores por nodo

5.2 Configuración de productos - <nombre>.qcr

Cada producto deberá tener un nombre asociado y el archivo de configuración de dicho producto tendrá de nombre el nombre del producto con la extensión *'.qcr'*

Descripción de las principales secciones:

1) [**Base**]
   - **INSTRUMENT_LIST**: lista de radares que serán procesados
   - **VARLIST**: lista de todas las variables que se desean procesar
   - **VARLISTMIN**: lista de las mínima cantidad de variables para empezar a procesar
   - **SAVE_NETCDF**: formato de archivos de producto (netcdf)
   - **NETCDF_OUTPATH**: directorio destino para los productos generados
   - **FILTERS**: lista de los filtros que se desea aplicar a los datos de entrada
   - **DATA_EXT**: posibles extensiones de los datos de radar a leer (H5, vol, nc)
   - **FILTERS_PLOT**: opción para realizar los gráficos de los filtros y variables corregidas (Bool)
   - **NO_RAIN_REF_VALUE**: valor de reflectividad que representa la no precipitación
   - **SAVE_KEEP_ORIGINAL**: si es verdadero se agregan campos adicionales corregidos al archivo de salida NetCDF. Si es falso, los campos corregidos se guardarán con los nombres originales

Servicio Meteorológico Nacional Argentina

2) [**Nombre Filtro**] En esta sección se definen todos los parámetros que serán utilizados por los filtros para procesar los datos de entrada. En Arruti y otros (2021) se detallaron algunos parámetros relevantes solo de los filtros que se encuentran activos al momento de generar la presente Nota Técnica.

5.3 Puesta en ejecución

Al momento de generar esta Nota Técnica, el sistema se encuentra corriendo en forma operativa-experimental cada 30 minutos (utilizando la aplicación *crontab* del SO) en el cluster HPC Huayra-Muyu (de Elía y otros 2020). Para su instalación se siguió la guía de instalación que se encuentra en el repositorio GitLab del SMN (https://gitlab.smn.gov.ar/ID/qc-radarv1.0/blob/master/README.md). Es requisito necesario contar con un entorno de conda con Python 3.6 o superior e que incluya los módulos listados en `qc_radar.yaml` que contiene las versiones de bibliotecas que conforman una instalación probada. Además, se debe contar con un compilador de Fortran para realizar la compilación con el archivo provisto en el repositorio.

6. REQUERIMIENTOS DE HARDWARE DEL SISTEMA

Para estimar los requerimientos de hardware necesarios para realizar el control de calidad de los datos de radar en un tiempo que pueda resultar de utilidad para los distintos usuarios se realizaron un conjunto de pruebas orientadas principalmente a estimar el nivel de paralelismo adecuado que minimice los tiempos de ejecución y medir la memoria requerida.

El procesamiento de cada volumen de radar está paralelizado en *threads* y una de las pruebas consiste en ir incrementando la cantidad de *threads* usados para procesar un volumen y medir el tiempo total. En la Figura 2a se muestran varias líneas correspondientes al tiempo total de procesar varios volúmenes en serie (se procesa un volumen a continuación del otro), mientras que en la Figura 2b todos los volúmenes se ejecutan simultáneamente.

En ambos casos se puede observar que utilizar más de 4 *threads* no mejora los tiempos de manera significativa. También se puede notar que aplicar el control de calidad de manera simultánea a los distintos radares disminuye considerablemente el tiempo total de ejecución. Además, durante las pruebas se estimó la memoria RAM necesaria para procesar cada volumen resultando en un promedio de 1.5 Gb por volumen de radar.

Es importante resaltar que los tiempos y memoria requerida para procesar un volumen de radar depende de la cantidad de datos meteorológicos y no meteorológicos que el radar esté detectando. Para estas pruebas, siempre se utilizó un único volumen que considera una situación con gran cobertura de datos y fenómenos que el sistema de control de calidad busca corregir (bloqueos, interferencias, atenuación, entre otros).

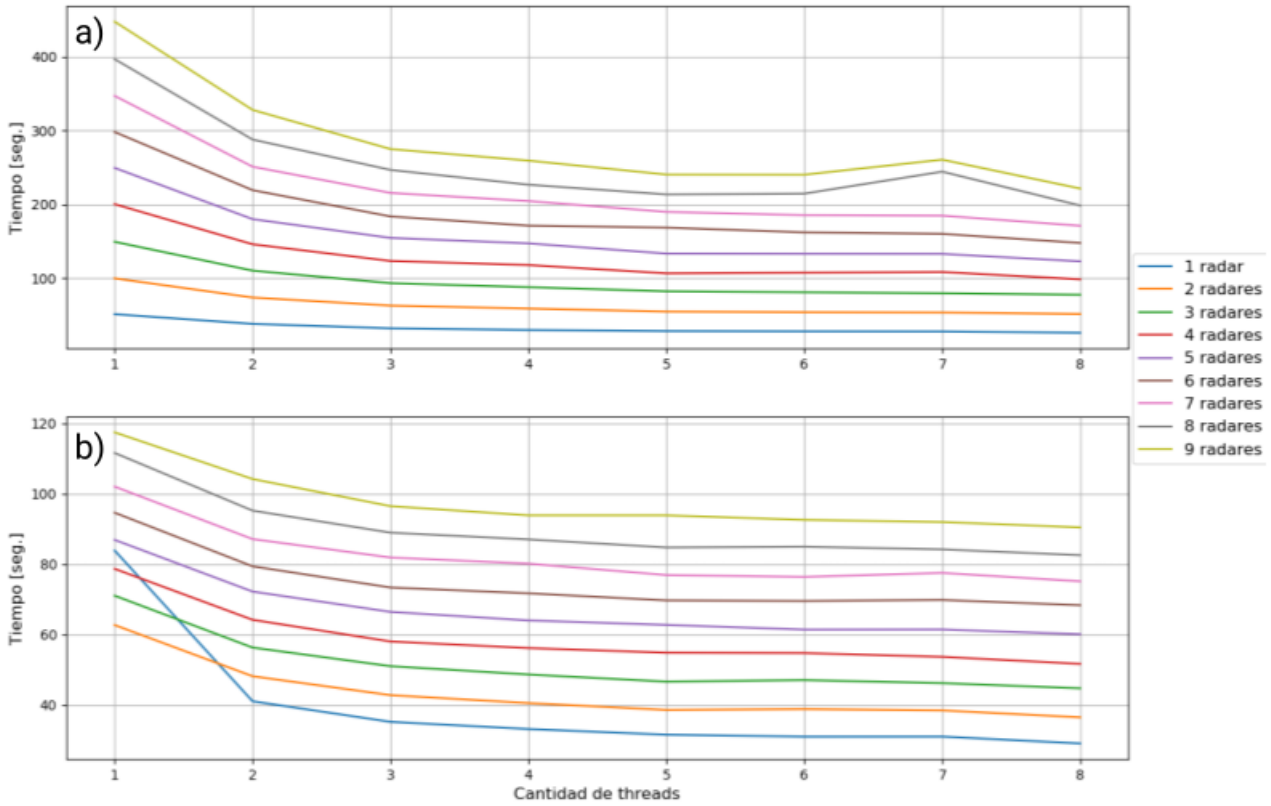

Fig. 2 *Tiempo de ejecución en función de la cantidad de threads utilizados para procesar distintas cantidades de radares en a) serie y en b) paralelo.*

REFERENCIAS

Amburn, S. A. y P. L. Wolf, 1997: Vil density as a hail indicator. Wea. and. Forecasting, 12, 473–478.

Anderson, M.E. L. Carey, W. Petersen y K. Knupp, 2011: C-band dual-polarimetric radar signatures of hail. Electron. J. Oper. Meteor.. 12. 1-30.

Arruti, A., P. Maldonado, M. Rugna, M. Sacco, J. Ruiz, L. Vidal, 2021: Sistema de control de calidad de datos de radar en el Servicio Meteorológico Nacional - Parte I: Descripción del algoritmo. Nota Técnica SMN 2021-86.

Auer, A. H., Jr, 1994: Hail recognition through the combined use of radar reflectivity and cloud-top temperatures. Mon. Wea. Rev., 122, 2218–2221.

de Elía R., L. Vidal, P. Lohigorry, R. Mezher y M. Rugna, 2017: La red Argentina de radares meteorológicos de Argentina. Nota Técnica SMN 2017-39.

de Elía R., M. Gené, V. Sala, P. Loyber, Y. García Skabar, M. Arianna, 2020: Un salto en la potencia de cálculo en el SMN: cómo se adquirió el nuevo HPC. Nota Técnica SMN 2020-67.

Servicio Meteorológico Nacional
Argentina

Instrucciones para publicar Notas Técnicas

En el SMN existieron y existen una importante cantidad de publicaciones periódicas dedicadas a informar a usuarios distintos aspectos de las actividades del servicio, en general asociados con observaciones o pronósticos meteorológicos.

Existe no obstante abundante material escrito de carácter técnico que no tiene un vehículo de comunicación adecuado ya que no se acomoda a las publicaciones arriba mencionadas ni es apropiado para revistas científicas. Este material, sin embargo, es fundamental para plasmar las actividades y desarrollos de la institución y que esta dé cuenta de su producción técnica. Es importante que las actividades de la institución puedan ser comprendidas con solo acercarse a sus diferentes publicaciones y la longitud de los documentos no debe ser un limitante.

Los interesados en transformar sus trabajos en Notas Técnicas pueden comunicarse con Ramón de Elía (rdelia@smn.gov.ar), Luciano Vidal (lvidal@smn.gov.ar) o Martin Rugna (mrugna@smn.gov.ar) de la Dirección Nacional de Ciencia e Innovación en Productos y Servicios, para obtener la plantilla WORD que sirve de modelo para la escritura de la Nota Técnica. Una vez armado el documento deben enviarlo en formato PDF a los correos antes mencionados. Antes del envío final los autores deben informarse del número de serie que le corresponde a su trabajo e incluirlo en la portada.

La versión digital de la Nota Técnica quedará publicada en el Repositorio Digital del Servicio Meteorológico Nacional. Cualquier consulta o duda al respecto, comunicarse con Melisa Acevedo (macevedo@smn.gov.ar).