Contract No.:

This manuscript has been authored by Battelle Savannah River Alliance (BSRA), LLC under Contract No. 89303321CEM000080 with the U.S. Department of Energy (DOE) Office of Environmental Management (EM).

Disclaimer:

The United States Government retains and the publisher, by accepting this article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for United States Government purposes.

Manuscript (non-LaTeX)
Click here to access/download;Manuscript (non-LaTeX);Article-Revised.docx

The Application of a Genetic Algorithm to the Optimization of a Mesoscale Model for Emergency Response

David Werth,[a] Robert Buckley[a]

[a] *Savannah River National Laboratory, Aiken,SC*

Corresponding author: David Werth, David.Werth@srnl.doe.gov

File generated with AMS Word template 1.0

ABSTRACT

Besides solving the equations of momentum, heat, and moisture transport on the model grid, mesoscale weather models must account for subgrid-scale processes that affect the resolved model variables. These are simulated with model parameterizations, which often rely on values preset by the user. Such 'free' model parameters, along with others set to initialize the model, are often poorly constrained, requiring that a user select each from a range of plausible values.

Finding the values to optimize any forecasting tool can be accomplished with a search algorithm, and one such process – the genetic algorithm (GA) - has become especially popular. As applied to modeling, GAs represent a Darwinian process – an ensemble of simulations is run with a different set of parameter values for each member, and the members subsequently judged to be most accurate are selected as 'parents' who pass their parameters onto a new generation.

At the Department of Energy's Savannah River Site in South Carolina, we are applying a GA to the Regional Atmospheric Modeling System (RAMS) mesoscale weather model, which supplies input to a model to simulate the dispersion of an airborne contaminant as part of the site's emergency response preparations. An ensemble of forecasts is run each day, weather data are used to 'score' the individual members of the ensemble, and the parameters from the best members are used for the next day's forecasts. As meteorological conditions change, the parameters change as well, maintaining a model configuration that is best adapted to atmospheric conditions.

File generated with AMS Word template 1.0

SIGNIFICANCE STATEMENT

We wanted to develop a forecasting system by which a weather model is run over the Savannah River Site each day and repeatedly adjusted according to how well it performed the previous day. To run the model, a series of values ('parameters') must be set to control how the model will calculate winds, temperatures, and other desired variables. Each day the model was run several times using different combinations of these parameters and later compared with observed meteorology. Parameters that produced the most accurate forecasts were preferentially reused to create the forecasts for the next day.

The process was tested for the summer of 2020 and exhibited lower errors than forecasts produced by the model using default values of the parameters.

File generated with AMS Word template 1.0

1. Introduction

Computer forecasting tools often require that a user preset a series of poorly constrained 'parameters' in the code that dictate how the model inputs relate to the output (the forecast), and the final result can be sensitive to the values to which the parameters are set. This holds true for a wide variety of forecasting algorithms (mesoscale models, analog forecasts, neural networks, etc.), and selecting these parameters can be as challenging as the problem the model is intended to solve.

The problem can be expressed in terms of a 'space' of N parameters (x_i; i=1, N), with each point in the space assigned an objective error E(**x**) based on the errors in a forecast created with those parameters. The goal then becomes to minimize the error by finding the point in the parameter space at which E(**x**) is at its lowest value. A sensitivity study, in which the model is run with a variety of values in hope of finding an optimal set (that is, sampling several points in the parameter space and selecting the one with lowest error) is time consuming and inefficient. An alternative solution is to apply a search algorithm – a systematic process to evaluate one set of points, then select a new set to evaluate based on the errors from the previous set. This often involves starting with a set of closely-spaced points, evaluating their respective errors (E(**x**)), estimating the local gradient of E(**x**), and moving the sampling points a small distance 'downhill', ideally toward the overall minimum in parameter space, before repeating the process (e.g., Duan et al., 2017; Severijns and Hazeleger, 2005; Werth et al., 2019).

In practice, this can take several forms. One in particular seeks to emulate the same Darwinian process involved in biological reproduction – genetic algorithms (GAs) (Whitley, 1994). GAs involve first encoding the parameter set as a string of values, reproducing the

File generated with AMS Word template 1.0

structure of genes on a chromosome (e.g., Lightner and Graham, 2006). (That is, each point in parameter space is represented as a linear sequence.) After an initial set of chromosomes is created and the fitness of each is determined, a new generation is created in a way similar to what happens in nature – the better performing chromosomes are paired-off and 'crossed' with each other, with a portion of genes from one 'parent' and a portion from the other combined to create a new 'child' chromosome. The addition of 'mutation' to this process – perturbing the child parameter values slightly – will help maintain the genetic diversity necessary for continual improvement.

GAs have been used to select the parameters for a wide variety of algorithms – finding the best parameters for a regional sea ice model (Sumata et al., 2019), the weights of a neural network model to predict seasonal rainfall (Elsanabary and Gan, 2014), the predictors and associated weights of an analog weather forecasting model (Horton et al., 2017), the distribution of unknown sources of an airborne tracer that comports with the concentrations recorded at a series of detectors (Haupt et al., 2006), and the best way to post-process the results from a weather model (Roebber, 2018; Bakhshaii and Stull, 2009). Sumata (2013) compared the benefits of a genetic algorithm with those of a gradient descent method (the method used by Duan et al. (2017), O'Steen and Werth (2009), and Werth et al. (2019)) for optimizing a sea ice model, and found the genetic algorithm to be superior. While this is for a single case, there are other benefits to using a genetic algorithm (specifically, working better when we expect the error surface $E(\mathbf{x})$ to be highly irregular) that recommend its use.

Mesoscale weather models often come with a long list of required settings, and the onus for selecting the most appropriate values is on the user, making them especially suited for GAs. One such model, the Regional Atmospheric Modeling System (RAMS; Pielke et al., 1992; Cotton et al., 2003), is used extensively at the Savannah River Site (SRS) for

File generated with AMS Word template 1.0

mesoscale weather forecasting and informing decision making in response to a toxic airborne release. This requires that the model be run over a large domain with a small grid spacing to resolve the small-scale wind variations that can have a large effect on airborne transport, with the results subsequently used to force a separate model of airborne dispersion. The latter simulation is relied on for its depiction of contaminant concentrations and surface deposition downwind of the source.

The SRS, located in South Carolina, is a Department of Energy (DOE) facility containing ongoing tritium and other nuclear work, as well as a remediation program for contaminated areas on site, and comprises approximately 800 square kilometers. To meet federal requirements, the site must maintain its airborne dispersion modeling capability to support emergency operations should an accident result in the release of a toxic or radioactive contaminant. Site operations also include the scheduling of outdoor work and informing decision making regarding controlled burns of forested areas by the Savannah River Forest Service, requiring weather forecasts. The site runs RAMS to both produce daily weather forecasts and, when required, provide meteorological input to a separate airborne dispersion model to support emergency response.

To optimize RAMS for use in dispersion modeling and emergency management, we apply a GA process to select parameter values – creating an ensemble of forecasts with various values of the model parameters, 'scoring' them according to their accuracy with respect to observations, and using the best members to create a new ensemble. We cannot accomplish this by optimizing the parameters for a particular time period of the past and assuming that this same parameter set will yield the lowest error for future forecasts - as weather systems come and go and the seasons change, the best parameters will also change. (In effect, the function E(**x**) will change from day to day, with the minimum occupying

File generated with AMS Word template 1.0

different points in the space.) To account for this, we can do the optimization by creating a new generation each day – running an ensemble for today, validating it when the observations for the day become available, and using the results to select parameters for a new ensemble tomorrow. The assumption is that the best parameters of today are likely to do well tomorrow, but new values are also needed in anticipation of changing conditions. Ideally, use of the GA method results in an ensemble of simulations that most accurately responds to the latest meteorological conditions in the region.

Our experiment comprises the optimization of a mesoscale model run daily and subsequently used to drive an airborne dispersion model. It begins in May of 2020 and continues through the summer. This season tends to be characterized less by the clearly-defined synoptic forcing systems that dominate in winter and more by small-scale forcing that mesoscale models often have difficulty simulating. Therefore, summer presents a unique challenge to the GA algorithm.

2. Mesoscale Atmospheric Model

The RAMS (v. 6.2.06, van den Heever (2020)) model is set to run over a 2-grid domain, centered at SRS (Fig. 1). The outer domain comprises 62 x 62 grid points, with 6 km grid spacing, and an inner domain of 62 x 62 grid points, with 1.5 km grid spacing. For both grids, the vertical spacing is 30 m in the lowest layer, increasing by a factor of 1.15 in each overlying layer, up to a maximum of 1000 m. The model is run with an 8 second time step for grid 1 and 2 seconds for grid 2.

The boundary conditions are from the 12 km North American Model (NAM) (NCEI, 2021) simulation that initiates daily at 0000 UTC (5 hours before the data are downloaded, see below) and is then integrated over 72 hours. The influence of the boundary conditions on

File generated with AMS Word template 1.0

148 the model dynamics is induced through a relaxation scheme (Clark and Hall, 1991) – a
149 relaxation term is added to the model prognostic equations such that they are nudged toward
150 the boundary values on timescales set by the user, which range from a smaller value (for
151 stronger forcing) at the outer edge to a larger value (for weaker forcing) at a distance of five
152 gridpoints from the edge (Saleeby, 2020). RAMS is set to run with the Harrington radiation
153 scheme (Harrington, 1997), allowing for the full interaction of incoming and outgoing
154 radiation with clouds. On grid 1, convection is parameterized with the Kuo scheme (Kuo,
155 1974), by which resolved instability is converted into convection. No convection
156 parameterization is run on the finer grid. On both grids, a prognostic cloud/precipitation
157 scheme is used (Saleeby and Cotton, 2004; 2008). Subgrid-scale diffusion is represented as a
158 flux-gradient process, by which fluxes are related to the resolved gradients (Stull, 1988).
159 Horizontal diffusion is accomplished with the horizontal deformation scheme of Smagorinsky
160 (1963), while vertical diffusion is calculated according to the Mellor-Yamada scheme (Mellor
161 and Yamada, 1982). The latter involves calculating a budget of unresolved turbulent kinetic
162 energy (TKE), which is then used to determine the eddy diffusivity – the rate at which the
163 flux rises with the gradient.

164 Fluxes between the land surface and the atmosphere are calculated with the Land
165 Ecosystem-Atmosphere Feedback (LEAF-3) land surface scheme (Walko et al., 2000; van
166 den Heever, 2020). This model uses United States Geological Survey (USGS) vegetation
167 data at 1 km horizontal resolution (developed with Advanced Very High Resolution Radar
168 (AVHRR) data from 1992 to 1993) to assign a set of subgrid-scale ‘patches’ to each model
169 grid cell and solves for the surface fluxes over each patch. The total flux into the overlying
170 atmospheric grid box is then calculated as a mean of the individual patch fluxes, weighted
171 according to patch area. To calculate fluxes at the ocean surface, the model uses a prescribed

File generated with AMS Word template 1.0

annual cycle of sea surface temperatures (SSTs), obtained from monthly-averages of climatological data at 1.0° horizontal grid resolution (Reynolds et al., 2002). Within the model, this data is interpolated between successive months to vary linearly from one time step to the next and create a smooth variability in SSTs. .

The model uses two topographic datasets – one at 30-s resolution (~833 m) for grid 1 and one at 6-s resolution (~167 m) for grid 2. For each grid, the topographic data exists at a finer resolution than the model grid spacing, and the model prepares the data by first mapping it to a coarser resolution grid (though not the same grid on which the model solution is subsequently calculated, Appendix A). On this intermediate grid, the model uses a 'silhouette' averaging scheme that calculates the average topographic height as seen from the south, the average topographic height as seen from the west, then averages those two values (Saleeby, 2020). A simple averaging is also done over the entire grid box, and the two averages are then averaged together to create an overall resolved topographic height.

The various schemes involve parameters that must be preset (Appendix A), with different values recommended based on resolution, season, and weather pattern. Other free parameters exist in the model, but we selected these nine because of their potential to greatly affect the model dynamics, the uncertainty in their actual values (e.g., soil moisture) and suggestions in the technical manual that users experiment with multiple values. The problem of selecting the optimal values of these parameters will be accomplished by the GA scheme as explained in the next section.

File generated with AMS Word template 1.0

3. The GA Process

RAMS is run as an ensemble of 10 members, each of which is executed with a different set of prescribed model parameters (Appendix A). To optimize these, we apply the GA algorithm – running the model with various parameter sets, scoring the resulting forecasts, then using the best-scoring parameters to create a new generation.

a. The GA Cycle

Many attempts to optimize a model involve running the same simulation repeatedly and adjusting the parameters after each generation (Werth et al., 2019, for example, ran multiple simulations of Oct. 23-Oct. 27 of 1994 over Europe, and applied a process to select better parameters after each). When run this way, the GA benefits from a ratchet effect – the improvements may eventually become small or stall out completely, but the model will never become worse. For our purposes, however, each generation will involve a unique scenario – the simulation of a new day. Our goal is to optimize the model by selecting the best performing parameters for one day, then applying a combination of those same parameters the next day. As the weather changes from day to day, the model will be run with parameters that are constantly adjusted, keeping the model optimized despite the changing conditions.

The process comprises the following steps, each executed in sequence:

i) Each night, starting at 1:00am local time (LT), the most recent NAM boundary condition data are loaded, and the RAMS model integration is started at 0600 UTC (2:00am LT), running over a 30-hour period. The first 6 hours of data are not used to allow for model spin up, resulting in ten 1200 UTC to 1200 UTC forecasts for the next day. The simulations are generally complete by 7:00am LT, and the results made available for emergency responders. Data are saved hourly.

File generated with AMS Word template 1.0

ii) At 11:30pm LT (0330 UTC), data collected during the preceding day (1200 UTC to 0300 UTC, 8:00am LT to 11:00 pm LT) are used to create a composite error 'score' for each of the 10 members of the ensemble (as described in Appendix B), using data from a set of meteorological towers located onsite and a nearby airport (Fig. 1b). The score (which we hope to minimize) is calculated as a weighted sum of the root mean square errors at specific levels for temperature, wind speed, wind direction and turbulent kinetic energy when the model results are compared to data from sensors at the 5 towers.

iii) Another algorithm compares the scores and creates a list of parameters for a new ensemble, applying a selection process to preferentially assign to this list the parameter values from the best-scoring members of the current ensemble. This list contains ten sets of 9 parameters each, one set for each of the ten new ensemble members.

iv) At 1:00am LT (1.5 hours after step ii takes place), a new ensemble is created using the values from this list, and the process is repeated.

b. Creating a New Generation

Following calculation of the ten scores, a new generation of ten members (#1 – #10) must be prepared, using the parameters from the current generation. In keeping with the assumption that the best parameters should not vary much over the timescale of one day, it is desired to reuse the best-performing parameters from the previous day for the current day's forecast. Therefore, before the GA is applied, the previous day's best member is assigned (unaltered) as Member #1 of the next generation, and the GA is subsequently used to create Members #2 – #10 (as described below).

To create the new chromosomes for the other new members, we apply a roulette wheel selection process as described by Jones (2002). We first create five 'couples' by randomly

File generated with AMS Word template 1.0

241 selecting members – each couple is created by drawing (with replacement) a member from
242 the ensemble (creating the first parent), then drawing a second (possibly the same) member
243 (creating the second parent) and pairing them. Each member is assigned a probability of
244 being chosen according to its score (Fig. 2). The probabilities range from 0.34 for the best
245 (lowest) scoring member to 0.034 for the worst (highest) scoring member. This makes it very
246 likely that the best member will be selected several times to serve as a parent (beyond its
247 assigned role as Member #1), but even the worst member has an approximately 1 in 4 chance
248 of being selected at least once after the nine draws[1].

249 The five couples are then each used to produce two children (except the first couple,
250 which produces only one child (#2) while the other child (#1) is the best member from
251 yesterday), using a one-point crossover process (Fig. 3). A crossover point is randomly
252 selected, and a child (C1 in Fig. 3) is created by concatenating genes on one side of the
253 crossover point from the first parent with genes from the second parent on the other side.
254 This will make the process of creating new members relatively conservative, keeping
255 together the parameters that worked well together and making rarer large differences between

[1] For a single draw, the probability of that member being selected is 0.034, leaving a 0.966 chance of not being selected. For that member to not be selected a single time after 9 draws, the probability is then (0.966)^9 = 0.732, leaving about a 25% chance that it will be selected at least once.

File generated with AMS Word template 1.0

256 the children and their parents. The process is repeated with a new crossover point to create a
257 second child for each couple (C2 in Fig. 3). In this way, a set of 9 new children (and one pre-
258 existing child) is created from the five couples and is ready to serve as the new ensemble.

259 As a final step, a mutation process is applied to members #2-#10 to introduce new
260 diversity in the population – each parameter value (P) has a 30% chance of being mutated,
261 and (if selected for mutation) is randomly perturbed by an amount (δ) that will reset it to a
262 new value within a prescribed range from its current value:

263 $P_{new} = P_{old} + \delta$

264 $$\delta = \mu \frac{P_{max} - P_{min}}{8}$$

265 where μ is a random number between -0.5 to 0.5., and P_{max} and P_{min} are the maximum
266 and minimum parameter values from Table 1.

267 Apart from the 10 members of the ensemble, an additional 'alternative' (Alt) member
268 is created each day, using standard values of the model parameters (Table 1). These were set
269 to either lie within the recommended range (e.g., TNUDLAT) or represent plausible values
270 (e.g., STGOFF). The Alt run is scored along with the other members and serves as a default
271 control for comparison to verify that the GA-produced forecasts are superior to the forecast
272 we could get by simply using a single run with typical, constant values of the parameters.

273

274

275

File generated with AMS Word template 1.0

c. The GA Experiment

We start the procedure by placing all members at the same point in the parameter space - the process was initiated at 1:00am LT on May 13, 2020 by running the 10-member ensemble using the default parameter values (Alt, Table 1) for each member (for 10 identical forecasts). The resulting forecast covers the period 1200 UTC May 13th to 1200 UTC May 14th. At 11:30pm LT on May 13th, these were all scored (yielding identical scores) and the GA randomly selected parents as described earlier. (Observations for only the first 15 hours of the 24-hr simulation are available for scoring (Appendix B)). With identical parents, the parameters were identical for all children until the mutation step. At that point, differences were introduced to yield a new generation that differed (slightly) from the previous generation. By repeating this process day after day, the differences grew, and the ensemble was forced further from its starting point. The experiment was run out to August 31st, 2020, with the results from May considered as spin-up. The results therefore comprise simulations run nightly from June-August, allowing us to (1) examine the summer forecasting skill of the GA, (2) explore the benefit of using an ensemble GA versus running a standard mesoscale model, and (3) determine how the model ensemble evolves over time as forced by meteorological conditions.

4. Results

For each generation the conditions will be slightly different from those that shaped the previous generation. While no single optimum exists, the GA should still have the effect of gradually steering the parameters to values that fit the current conditions. As the summer progressed, for example, simulations with wetter soil (Fig. 4a) were favored, with the optimum soil temperature cooling until the 40th generation (June 22nd) before rising again

File generated with AMS Word template 1.0

(Fig. 4b) to generation 60, falling to generation 80, and rising thereafter. The two topography-related parameters – TOPTENH and TOPTWVL (Appendix A) – experienced long-term decreasing and increasing trends, respectively (Fig. 4c, d), despite the fact that topography itself is constant. Such adjustments in the prescribed parameters can compensate for errors in other parts of the model (e.g., Ruckstuhl and Janic, 2020). For example, the topographic forcing in the Alt run appeared to be of too fine a scale, and the forecasts benefitted from coarser topography – when TOPTENH is low, the silhouette averaging is decreased (for increased topographic smoothing), and when TOPTWVL is high, the minimum topographic wavelength is increased (also increasing topographic smoothing). The process also favors larger values of Lm (the length scale multiplier, Appendix A) (Fig. 4e), increasing the amount of parameterized vertical mixing. The fairly steady trends in Fig. 4 imply that the 'new' optimum is usually not far from the 'old' optimum, and that our procedure for sampling new points is keeping the model parameters optimized.

The prescribed mutation rate will dictate the 'distance' from the current best member that the algorithm searches to create new members. The assumption is that the better values will likely be close to the current value, but allowing for the possibility of better values lying further away. The algorithm can be assessed by how often the various new values lead to improvements – if the winning parameters tend to be the most distant, the search area is too small. Conversely, an excessively large search area will result in the current values being disproportionately reselected as the other members sample too-distant areas of the parameter space.

We can evaluate how well the algorithm is sampling the parameter space by comparing the changes in winning value to the overall changes – if the distributions are similar, we can be more confident that the GA is preferentially sampling better values while avoiding worse

File generated with AMS Word template 1.0

ones. The range of changes in winning parameter value (the red dots in Fig. 5, left) indicates that values of 0 (that is, instances in which reusing yesterday's value works best) are most common (counted over all parameters, changes of 0 occur in 37% of all cases), with winning perturbations further from 0 occurring less often. The winning values are likely to lie within one standard deviation of the mean of all perturbed values, and less likely to lie further out, with a frequency similar to what we would expect from a Gaussian distribution (Fig. 5, right). The probability of a new value being selected falls off with distance from the current value – smaller changes in the parameters are more likely to be best, with larger deviations from the current value being less likely to do well. The latter occurs often enough, however, to justify the inclusion of large deviations when selecting new values. We are therefore confident that the sampled space is adequately discovering newer, better values at a reasonable rate.

Besides providing an ensemble, the expected benefit of the GA is to improve the forecasts relative to what we can get from simply using a single run of the same model with constant parameters set to plausible values. To quantify this, we can compare the scores of the 10 members at each generation to that of the Alt run. Without a fixed target, we do not expect the ensemble to steadily and irreversibly outperform the Alt run, as changing conditions mean that the best members of one day may perform poorly the next. It is therefore possible that the Alt run will be better than at least a few of the ensemble members, and perhaps even better than the member that did best the previous day (which is automatically assigned a place in the next day's generation). After the first few generations, however, we expect that the ensemble will eventually outperform the Alt run.

For the first six generations (during spin up) (Fig. 6), the ensemble is only slightly better than the Alt run. For the six generations after that, the ensemble scores about as well as, and often worse than, the Alt run. For the month of June (generations 19-48), however, the

File generated with AMS Word template 1.0

ensemble does much better, scoring substantially lower than the Alt run, with a downward trend in the error difference. Generations in which all members do better (all points, not just the best, are below the zero line) are now common, with worse-scoring members (above the zero line) becoming rarer. In July (generations 49-79), the Alt member rarely outscores the GA members, and the same is true for much of August (generations 80-110). Near the end of the summer, however, the difference in scores tends to be smaller, with more frequent days with the Alt member outscoring GA members (albeit these are still in the minority).

A change point analysis was applied to define three periods when these values were well above the mean (Generations 1-38), well below the mean (Generations 39-86), and slightly above (Generations 87-107), and a trendline for each is overlaid onto Fig. 6. Values fall during the first period, followed by a period of weaker decline. Values then rise sharply during the third period as the summer ends and the GA does less well.

On June 30th (Generation 48), the error differences are in about the mid-range of all values (Fig. 6), so we show the time series for two towers from that day as being typical (Fig. 7). (Note that all towers were used to calculate the final score.) The ensemble and the Alt run tend to be similar except for larger swings in the latter, which contribute to its worse overall score. The Alt daytime temperatures are too high (Fig. 7a), the wind shifts to the east too early and accelerates late in the day (Fig. 7b, c). The Alt run TKE is more accurate for this day than the ensemble, but both suffer from a negative bias (Fig. 7d), a common occurrence during the multiple simulations.

At the end of the summer, we see two days (Aug. 25th, Aug. 30th) during which the Alt member outscores all members of the GA ensemble. On the latter day (also at two towers), the afternoon cooling is missed by the GA member, but captured in part by the Alt member (Fig. 8a). The Alt member also has more accurate winds during this time (Fig. 8b), and more

File generated with AMS Word template 1.0

closely tracks the observed wind speeds (Fig. 8c). While both model runs fail to produce the observed TKE (Fig. 8d), the Alt run's slightly higher values also contribute to its lower error score. On these two dates, the range of values for the ensembles shows little disagreement for forecasted temperature and TKE, with larger differences for wind direction and wind speed.

The benefit of the ensemble can be seen in its effect of reducing the model bias, calculated as the difference in the daily maximum value between the simulation and observations for each generation. Values are calculated for the Alt run and for each ensemble member, and the GA ensemble bias is calculated as an average of the biases of the 10 members. The bias in CL tower temperature at 18 m is substantially larger for the Alt run for all three months (Fig. 9a, a problem likely mitigated in the GA ensemble by the reductions in initial soil temperature), as it is for the wind speed at the same location (Fig. 9b). For TKE, the benefit is much smaller, with small differences between the ensemble and Alt runs (Fig. 9c). This is likely related to the large bias in TKE that persisted over the generations.

During an emergency response incident (or exercise), emergency managers will need projections of where and when the concentrations are expected to exceed prescribed thresholds. They can also benefit from knowing the estimated uncertainty in the projections, a commonly cited benefit of ensemble modeling. To demonstrate this, we create two such ensembles of 10 transport simulations each – one starting at 1200 UTC on June 23rd, and the other starting at 1200 UTC on July 14th. Each ensemble is run using meteorology from the 10 RAMS simulations from the respective days to force the Hybrid Single Particle Lagrangian Integrated Trajectory Model (HYSPLIT, Stein et al. 2015a, 2015b), a dispersion model designed to simulate the motions of individual particles in response to the input wind

File generated with AMS Word template 1.0

394 field. For each transport simulation, a continuous 1 Ci/hr release from the central SRS (Fig.
395 1b), starting at 1200 UTC (on June 23 and July 14, respectively) is assumed.

396 On June 23rd, the wind was primarily from the southwest, resulting in a plume extending
397 towards the northeast (Fig. 10, top). The ensemble members (the shaded contours) tend to
398 agree about the plume position – the positions of the 10^{-12} contours (the white lines) for the
399 different members lie relatively close to one another. On July 14th, the wind was out of the
400 west, but the meteorological ensemble was much more dispersive, resulting in a greater
401 spread of plumes among the ensemble members. The ensemble-averaged concentrations are
402 therefore spread over a much wider area, resulting in lower concentrations, particularly for
403 locations beyond the SRS boundary (Fig. 10, bottom). Also, the 10^{-12} contours now show
404 wide variability, with the members disagreeing about their location.

405 Site emergency managers require that the modeling results be presented as a series of
406 predefined metrics, allowing for consistent and rapid decision making. Two of those metrics
407 are defined along an arc centered at the source – the peak concentration (centerline
408 maximum) and the distance between points at which the concentration falls to one half of the
409 maximum (Full Width at Half Maximum (FWHM)). Depending on the day, the variability in
410 peak concentration among the members can be small (Figs. 11,12, top) or large (Figs. 11,12,
411 bottom), something also seen in the FWHM values (Figs. 13,14). Besides providing a
412 measure of confidence in the predictions, these differences among the members can allow the
413 emergency manager to consider a best-case scenario, a worst-case scenario, a spread of
414 possibilities, and a most likely scenario (the ensemble mean) when determining what
415 protective actions to take.

416

417

File generated with AMS Word template 1.0

5. Discussion and Conclusions

We applied a genetic algorithm to optimize the parameter values used by a mesoscale forecast model. The GA operated by creating forecasts with the mesoscale model, scoring them against observations, and selecting parameters from the better scoring ones to create forecasts for a subsequent period. The algorithm demonstrated its benefit by performing well when rated against forecasts produced with a set of unvarying parameter values, with lower error scores and a lower bias. The benefit mostly accrued to improvements in temperature, wind speed and direction, with a large negative bias in TKE persisting throughout the experiment. The process was not perfect – on occasion, the parameters that scored well on one day scored very poorly the following day – but these were relatively rare, so we assert that the GA process represented an improvement over the control (Alt) forecasts using constant parameter inputs. Although only tested for summer, the same processes can operate at any time of the year. The algorithm has been run for winter, but with a weaker improvement over the Alt runs than for summer. Certain times of the year are more amenable to successful prediction by a mesoscale model than others, and it is during the most challenging times when the genetic algorithm should prove most beneficial. The Alt scores generally rise (that is, get worse) during the summer, peaking in early August, then going on a downward trend that continues into the fall (not shown), so there is less room for improvement by the GA (possibly explaining the worse scores at the end of August in Fig. 6). This implies that summer indeed represents a difficult forecasting environment for a mesoscale model, with a greater possibility for improvement using the genetic algorithm.

The benefit can be seen in the improvement over the Alt runs. It could be asked if the overall changes in parameter values are simply compensation errors, and that the parameter values from the final generation would have worked well had they been used every day.

File generated with AMS Word template 1.0

Compensation errors would explain the drift in parameters had the GA steadily moved each parameter from its initial point to another, final point, from which it did not vary thereafter. This was not the case, however – even when the parameters tended to move in one direction overall (e.g., Fig. 4a), there was enough variability to suggest that they were responding to changing conditions and not simply being monotonically pulled to a best, final value.

As weather conditions change from one day to the next, the optimum will change as well, and the GA is designed to follow that optimum by favoring the members that are closest to it (i.e., score best). This is based on the assumption that the optimum would move steadily in the parameter space, and the GA would be able to 'follow' it. This assumption can be tested, however - Figure 5 suggests that the new optimum is usually close to the old one. Small (or no) changes in parameter values are favored, with only occasional instances in which large changes yield better scores. Therefore, we believe that the assumption is justified – if the optimum were instead subject to large jumps in random directions, the winning parameters would also tend to vary randomly, with no steady progression.

Despite the overall benefit, there are some disadvantages to the system as developed. For example, our scoring system used mostly daytime data (from 8:00am to 11:00pm), before much nighttime data has been collected. This means that during the night, when winds are weaker and harder to simulate, model errors will not be subjected to the selection process, allowing them to persist. The forecasting algorithm, however, was designed to approximate what an operational system would require – having the forecast ready by an early morning deadline – which required beginning the process of scoring the current day's forecasts and starting the model runs for tomorrow in the evening. The focus of the project is on site operations, which are conducted mostly during the day, so we believe the system as set up is of great benefit.

File generated with AMS Word template 1.0

Some trial and error was involved in selecting the values that control the GA process - the range of permissible parameter values, the mutation rate, and the weighting scheme of the error function, and a set for one period may not be optimal for another. Overall, however, we were satisfied that our values led to the generation of new members that efficiently explored the parameter space – consistently finding new values that maintained a lower error – during the experimental period. This process could eventually reach a point where this is no longer true - the improvement over the Alt run was highest for July (Fig. 6), but with a smaller (albeit positive) benefit for June and August. In an operational setting, forecasters could adjust the GA values (e.g., the mutation rate) as needed (as was done in O'Steen and Werth (2009) and Werth et al., 2019) but we wanted to avoid such manipulation for our experiment, allowing for a cleaner comparison of the different results.

The TKE is strongly affected by the length scale multiplier, which was nudged toward larger values during the period (Fig. 4e). Although the GA produced a small reduction in the negative bias relative to the Alt run, the GA-supported RAMS model produced TKE values that remained much too low. Presumably, larger values of the multiplier would have increased TKE further and reduced the negative TKE bias, but this would likely have been at the price of excessive vertical mixing, which would have reduced vertical gradients in wind and temperature and increased the errors in wind speed and direction. This highlights the difficulty of selecting a composite error - our weighting scheme penalized wind errors more heavily, which allowed TKE errors to persist, but this was considered the best option for optimizing the input to a dispersion model. Part of the goal was to challenge the algorithm by charging it with optimizing a suite of model outputs, all of which we desire to be accurate. These can often come into conflict – improving one could degrade another – and we believe our study does reveal the difficulty in optimizing a complete mesoscale simulation.

File generated with AMS Word template 1.0

The ensemble of simulations serves as input for an ensemble of Lagranian airborne dispersion simulations, which are used to inform the decision-making process in the event of an accidental release. The variability in the HYSPLIT simulations can be a useful indicator of the range of possible outcomes of such a release, possibly by including areas within the danger zone that would have been designated as safe by a single model run. In this regard, the GA should be a useful addition to the SRS emergency management organization.

Acknowledgments.

Stephen Noble of SRNL created the graphics for Figure 1.

Data Availability Statement.

Data provided on request.

File generated with AMS Word template 1.0

512 APPENDIX A

513 **Selected Parameters**

514 Each parent is represented as a single chromosome that comprises nine genes, each of
515 which represents a model parameter value. The nine model parameters adjusted by the GA
516 are listed in Table 1, along with their initial values (which are set to the Alt values), and are
517 the same ones used in Werth et al. (2019).

518 The parameters are described as follows:

519 AKMIN, Grid 1: This parameter is used to set a minimum value of K – the
520 coefficient of proportionality in the flux-gradient parameterization. For example, the zonal
521 heat flux due to unresolved eddies is represented as:

522 $[u'T'] = -K \ \Delta[T]/\Delta x$

523 in which a bracket represents the resolved variable. In the horizontal, K is determined by
524 the deformation scheme. To avoid excessive small-scale noise from accumulating in the
525 domain, however, a minimum threshold (AKMIN) is preset.

526 TNUDLAT (seconds): On each grid, the model imposes the effect of the boundary
527 conditions through a Newtonian relaxation scheme – nudging the variables toward the
528 boundary values by an amount proportional to the difference between the prescribed
529 boundary values and the model values. The strength of the nudging is adjusted by the
530 relaxation timescale (in seconds), with smaller values denoting a stronger effect. TNUDLAT
531 represents the nudging timescale applied to both grids at the outer boundary.

532 TNUDCENT (seconds): As for TNUDLAT, but for the interior of the domain.
533 Moving from the domain edge to 5 grid points inward, the nudging timescale rises

File generated with AMS Word template 1.0

parabolically from TNUDLAT to TNUDCENT, and it uses the latter value throughout the interior.

SLMSTR: The initial soil moisture, expressed as a fraction of total saturation.

STGOFF (°C): Soil temperature offset. The model soil temperature at all levels is initialized with a value calculated by adding the temperature of the lowest atmospheric layer to STGOFF.

AKMIN, Grid 2: As for Grid 1, but applied to the inner grid.

TOPTENH: The high-resolution topographic data are first averaged over an intermediate grid of resolution coarser than the model grid (and dictated by the model parameter TOPTWVL, see below), both for a silhouette average and a conventional average. TOPTENH is the weight assigned to the silhouette average when the silhouette and the conventional averages are combined to smooth the fine scale topography to the coarser resolution intermediate grid.

TOPTWVL: The averaging of the silhouette and conventional averages is done on the intermediate grid. To avoid generating noise at the lower boundary, this grid is of a resolution coarser (by a factor of TOPWVL/2) than the final model grid to which it is ultimately interpolated. Larger values of TOPTWVL will mean more smoothing of the topography.

Lm: The Mellor-Yamada scheme uses a prognostic equation to calculate resolved TKE, and this is subsequently used to calculate the vertical eddy diffusivity. This requires the calculation of a turbulent length scale L, and Lm is a multiplier to increase the turbulent length scale, thereby altering the vertical turbulent mixing.

File generated with AMS Word template 1.0

556 The parameters are ordered mostly as they appear in the model's input text file but similar
557 ones tend to be grouped together – relaxation times (#2, #3), topographic parameters (#7, #8)
558 and soil initialization (#4, #5). This means that some parameters will tend to vary together
559 (that is, be 'linked', as are genes on a chromosome). We do separate the two AKMIN values
560 (#1 and #6), allowing them to vary separately.

561 APPENDIX B

562 **Member Scoring**

563 Data are collected from a set of four meteorological towers on site, designated as A,
564 D, P, and CL (for CLimatology) (Fig. 1b), as well as from an airport near Augusta, GA
565 (Daniel Field, DF). Among other variables, all site towers and the airport tower collect
566 readings for temperature (T, °C), wind speed (S, m/s), wind direction (Dir, degrees), and
567 three of the site towers (A, D, P) collect data for turbulent kinetic energy (E, m2/s2). These
568 variables are crucial to airborne dispersion and serve as the input data for the site's dispersion
569 models. All site towers collect data at 61 m, while CL collects data at 18 m and 36 m as well.
570 The DF tower collects data from two levels – wind at 10 m and temperature at 2 m. At the
571 validation time (11:30pm LT), data from the towers exist for the period from 1200 UTC
572 (8:00am LT) to 0300 UTC the following day (11:00pm LT that same night). (TKE, however,
573 is only available until 0200 UTC.) Given that dispersion is sensitive to small-scale variations
574 in wind, using towers from across the domain to score the members is an appropriate way to
575 define an optimized model, and sets a higher bar for the genetic algorithm.

576 At validation, the data from the towers are collated, and data from each of the 10 RAMS
577 simulations are extracted at the points that coincide with the tower sensor locations, using a
578 bi-linear interpolation to fit the data from the model grid points to each tower. The
579 comparison is done at each tower over the N_k levels at which data are collected and at the N_t

File generated with AMS Word template 1.0

observation times that coincide with the hourly model output (i.e., the observed value at 1400 UTC at the 36 m level of the CL tower is compared to the model value extracted at that same time, level and location). At each tower, a score is calculated for each member m (S(m)) as the weighted root mean square error (RMSE) of the four variables:

$$S(m) = w_T RMSE_T + w_{Dir} RMSE_{Dir} + w_S RMSE_S + w_E RMSE_E$$

$$RMSE_x = \frac{1}{N_t N_k} \sqrt{\sum_{t,k}^{N_t,N_k} (x(t,k) - x_{obs}(t,k))^2}$$

The weight values are set as w_T=0.15, w_{Dir}=0.04, w_S=0.37, w_E=0.44, which assigns the same error to a 1°C error in temperature, a 3.75° error in wind direction, a 0.41m/s error in wind speed, and a 0.34 m2/s2 error in TKE. (For wind direction, we use the minimum absolute difference, which can never exceed 180°, so an observed wind at 350° and a model wind at 10° is counted as a difference of 20°.) A composite score is then calculated by summing the individual tower scores.

As with most attempts to optimize a model with a genetic algorithm, choosing the way the members are evaluated requires some subjective decisions as to how to create the objective error function. In our case, we wanted to give primacy to errors in wind direction, followed by errors in wind speed, TKE, and temperature, and the weight values were set accordingly. The weight values were set prior to the simulation period following a test simulation (on May 12th). Over the summer, the scores were made up of about 54% errors in wind direction, 22% of wind speed, 15% TKE and 9% temperature, so we are satisfied that the weighting scheme was having the desired effect of penalizing direction errors most heavily but stll applying selective pressure to the other errors.

File generated with AMS Word template 1.0

603 REFERENCES

604 Bakhshaii, A., and R. Stull, 2009: Deterministic Ensemble Forecasts Using Gene-
605 Expression Programming. *Wea. Forecasting*, **24**, 1431–1451,
606 https://doi.org/10.1175/2009WAF2222192.1.

607 Clark, T., and W. Hall, 1991: Multi-Domain Simulations of the Time Dependent Navier-
608 Stokes Equations: Benchmark Error Analysis of Some Nesting Procedures. *J. Comp. Physics*,
609 **92**, 456-481, https://doi.org/10.1016/0021-9991(91)90218-A

610 Cotton, W.R., R.A. Pielke, Sr., R.L. Walko, G.E. Liston, C.J. Tremback, H. Jiang, R.L.
611 McAnelly, J.Y. Harrington, M.E. Nicholls, G.G. Carrio, J.P. McFadden, 2003: RAMS 2001:
612 Current Status and Future Directions. *Meteor. Atmos. Physics*, **82**, 5-29.

613 Duan, Q., and Coauthors, 2017: Automatic Model Calibration: A New Way to Improve
614 Numerical Weather Forecasting. *Bull. Amer. Meteor. Soc*., **98**, 959–970,
615 https://doi.org/10.1175/BAMS-D-15-00104.1.

616 Elsanabary, M. H., and T. Y. Gan, 2014: Wavelet Analysis of Seasonal Rainfall
617 Variability of the Upper Blue Nile Basin, Its Teleconnection to Global Sea Surface
618 Temperature, and Its Forecasting by an Artificial Neural Network. *Mon. Wea. Rev*., **142**,
619 1771–1791, https://doi.org/10.1175/MWR-D-13-00085.1.

620 Harrington, J., 1997: The Effects of Radiative and Microphysical Processes on Simulated
621 Warm and Transition Season Arctic Stratus. Department of Atmospheric Science Bluebook
622 637, Colorado State University, 289 pp.

623 Haupt, S. E., G. S. Young, and C. T. Allen, 2006: Validation of a Receptor–Dispersion
624 Model Coupled with a Genetic Algorithm Using Synthetic Data. *J. Appl. Meteor. Climatol*.,
625 **45**, 476–490, https://doi.org/10.1175/JAM2359.1.

626 Horton, P., M. Jaboyedoff, and C. Obled, 2017: Global Optimization of an Analog
627 Method by Means of Genetic Algorithms. *Mon. Wea. Rev.*, **145**, 1275–1294,
628 https://doi.org/10.1175/MWR-D-16-0093.1

629 Jones, G., 2002: Genetic and Evolutionary Algorithms. Encyclopedia of Computational
630 Chemistry. 2

631 Kuo, H. L., 1974: Further Studies of the Parameterization of the Influence of Cumulus
632 Convection on Large-Scale Flow. *J. Atmos. Sci.*, **31**, 1232–1240,
633 https://doi.org/10.1175/1520-0469(1974)031<1232:FSOTPO>2.0.CO;2

634 Lightner, C., and J. Graham, 2006: A Heuristic Approach for Locating EMS Facilities
635 and Vehicles, Proceedings of the International Conference on Parallel and Distributed
636 Processing Techniques and Applications & Conference on Real-Time Computing Systems
637 and Applications. PDPTA 2006, Las Vegas, Nevada, USA, June 26-29, 2006, Volume 1

638 Mellor, G. L., and T. Yamada, 1982: Development of a Turbulent Closure Model for
639 Geophysical Fluid Problems. *Rev. Geophys. Space Phys.*, **20**, 851–875,
640 https://doi.org/10.1029/RG020i004p00851

641 NCEI, 2021: National Oceanic and Atmospheric Administration's National Centers for
642 Environmental Information: North American Mesoscale Forecast System (NAM).
643 https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/north-american-
644 mesoscale-forecast-system-nam

645 O'Steen, L., and D. Werth, 2009: The Application of an Evolutionary Algorithm to the
646 Optimization of a Mesoscale Meteorological Model, *J. Appl. Meteor. Climatol.*, **48**, 317–329

Pielke, R.A., W.R. Cotton, R.L. Walko, C.J. Tremback, W.A. Lyons, L.D. Grasso, M.E. Nicholls, M.D. Moran, D.A. Wesley, T.J. Lee, and J.H. Copeland, 1992: A Comprehensive Meteorological Modeling System – RAMS. *Meteorol. Atmos. Phys*., **49**, 69-91

Reynolds, R. W., N. A. Rayner, T. M. Smith, D. C. Stokes, and W. Wang, 2002: An Improved In Situ and Satellite SST Analysis for Climate. *J. Climate*, **15**, 1609–1625, https://doi.org/10.1175/1520-0442(2002)015<1609:AIISAS>2.0.CO;2

Roebber, P. J., 2018: Using Evolutionary Programming to Add Deterministic and Probabilistic Skill to Spatial Model Forecasts. *Mon. Wea. Rev*., **146**, 2525–2540, https://doi.org/10.1175/MWR-D-17-0272.1.

Ruckstuhl, Y., & Janjić, T., 2020: Combined State-Parameter Estimation with the LETKF for Convective-Scale Weather Forecasting, *Mon. Wea. Rev*., **148(4)**, 1607-1628.

Saleeby, S., 2020: CSU – RAMS, RAMSIN Model Namelist Parameters. 36 pp., https://vandenheever.atmos.colostate.edu/vdhpage/rams/rams_docs.php

Saleeby, S.M. and W.R. Cotton, 2004: A Large-Droplet Mode and Prognostic Number Concentration of Cloud Droplets in the Colorado State University Regional Atmospheric Modeling System (RAMS). Part I: Module Descriptions and Supercell Test Simulations. *J. Appl. Meteor*., **43**, 182-195

Saleeby, S.M. and W.R. Cotton, 2008: A Binned Approach to Cloud-Droplet Riming Implemented in a Bulk Microphysics Model. *J. Appl. Meteor. Climatol*., **47**, 694-703.

Severijns, C. A., and W. Hazeleger, 2005: Optimizing Parameters in an Atmospheric General Circulation Model. *J. Climate*, **18**, 3527–3535, https://doi.org/10.1175/JCLI3430.1

668 Smagorinsky, J., 1963: General Circulation Experiments with the Primitive Equations.
669 Part I, The Basic Experiment. *Mon. Wea. Rev*., **91**, 99–164, https://doi.org/10.1175/1520-
670 0493(1963)091<0099:GCEWTP>2.3.CO;2

671 Stein, A.F., R.R.Draxler, G.D. Rolph, B.J. B. Stunder, M. D,. Cohen, and F. Ngan, 2015a,
672 NOAA's HYSPLIT, Atmospheric Transport and Dispersion Modeling System. *Bul. Amer.*
673 *Meteor. Soc*., **96 (12)**, 2059-2077

674 Stein, A.F., R.R.Draxler, G.D. Rolph, B.J. B. Stunder, M. D,. Cohen, and F. Ngan, 2015b,
675 NOAA's HYSPLIT, Atmospheric Transport and Dispersion Modeling System (Supplement),
676 *Bul. Amer. Meteor. Soc*., **96 (12)**, ES203-ES207

677 Stull, R., 1988: An Introduction to Boundary Layer Meteorology, Kluwer Academic
678 Publishers, 666pp.

679 Sumata, H., Kauker, F., Gerdes, R., Köberle, C., and Karcher, M., 2013: A comparison
680 between gradient descent and stochastic approaches for parameter optimization of a sea ice
681 model, *Ocean Sci*., **9**, 609–630, https://doi.org/10.5194/os-9-609-2013

682 Sumata, H., F. Kauker, M. Karcher, and R. Gerdes, 2019: Simultaneous Parameter
683 Optimization of an Arctic Sea Ice–Ocean Model by a Genetic Algorithm. *Mon. Wea. Rev*.,
684 **147**, 1899–1926, https://doi.org/10.1175/MWR-D-18-0360.1.

685 van den Heever, S.C., Regional Atmospheric Modeling System (RAMS) Model
686 Documentation. 2020:
687 https://vandenheever.atmos.colostate.edu/vdhpage/rams/rams_docs.php.

688 Walko, R., and Coauthors, 2000: Coupled Atmosphere–Biophysics–Hydrology Models
689 for Environmental Modeling. *J. Appl. Meteor*., **39**, 931–944, https://doi.org/10.1175/1520-
690 0450(2000)039<0931:CABHMF>2.0.CO;2

691 Werth, D., G. Maze, R. Buckley, and S. Chiswell, 2019: The Application of an
692 Evolutionary Programming Process to a Simulation of the ETEX Large-Scale Airborne
693 Dispersion Experiment. *J. Appl. Meteor. Climatol*., **58**, 511–525,
694 https://doi.org/10.1175/JAMC-D-18-0098.1.

695 Whitley, D., 1994: A Genetic Algorithm Tutorial. *Stat. Comput*., **4**, 65–85,
696 https://doi.org/10.1007/BF00175354

697

File generated with AMS Word template 1.0

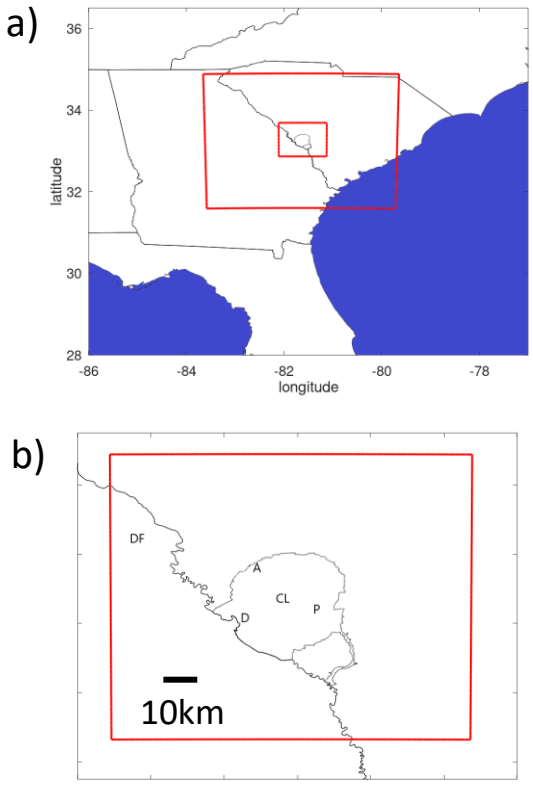
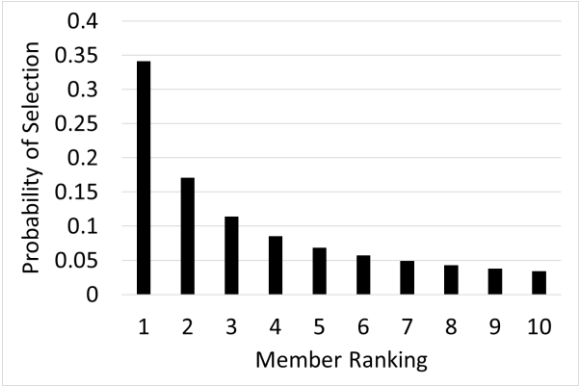


Figure 1 a) RAMS domain, with the two grids indicated by red squares, and b) close up of grid 2, map of the SRS, with the towers (A, D, P, CL) and airport (DF) indicated.



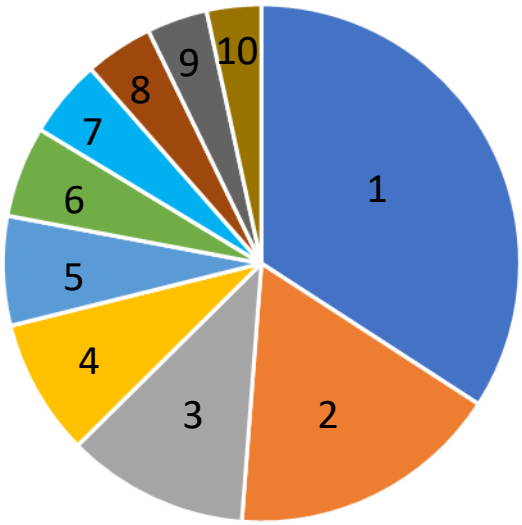


Figure 2 a) Distribution of probability assigned to each member, according to its ranking, depicted as a bar plot (top) and as a roulette wheel (bottom).

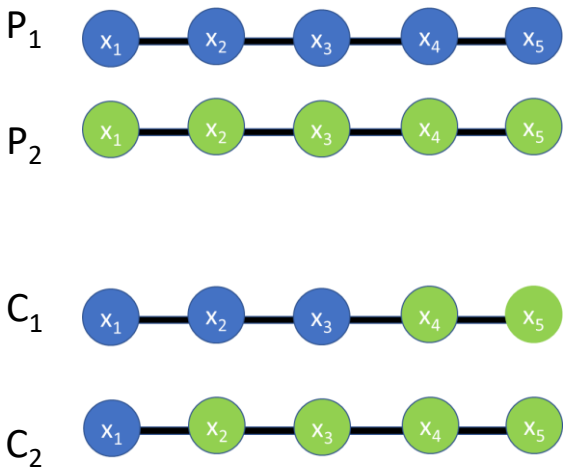


Figure 3 Schematic of two parent (P) chromosomes (top), and the two child (C) chromosomes (bottom), after crossover.

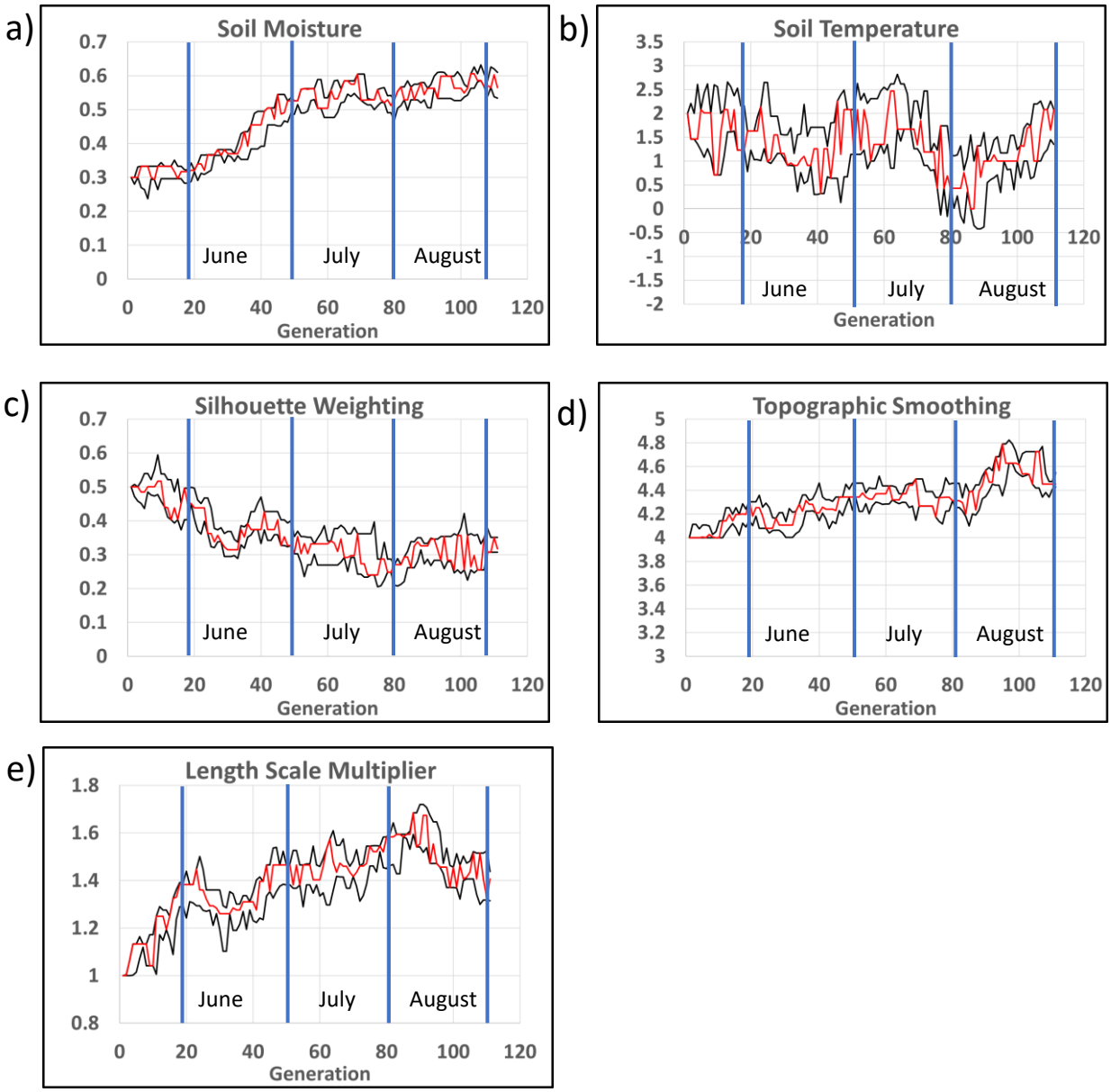


Figure 4 Range of parameter values (black lines) along with the 'winning' value (red line) for each generation for a) soil moisture (SLMSTR), b) soil temperature (STGOFF), c) silhouette weighting (TOPTENH), d) topographic smoothing (TOPTWVL), and e) length scale multiplier (Lm). (All are dimensionless except for soil temperature, which is in °C). The vertical blue lines indicate the first day of each month, with the series ending on September 1st.

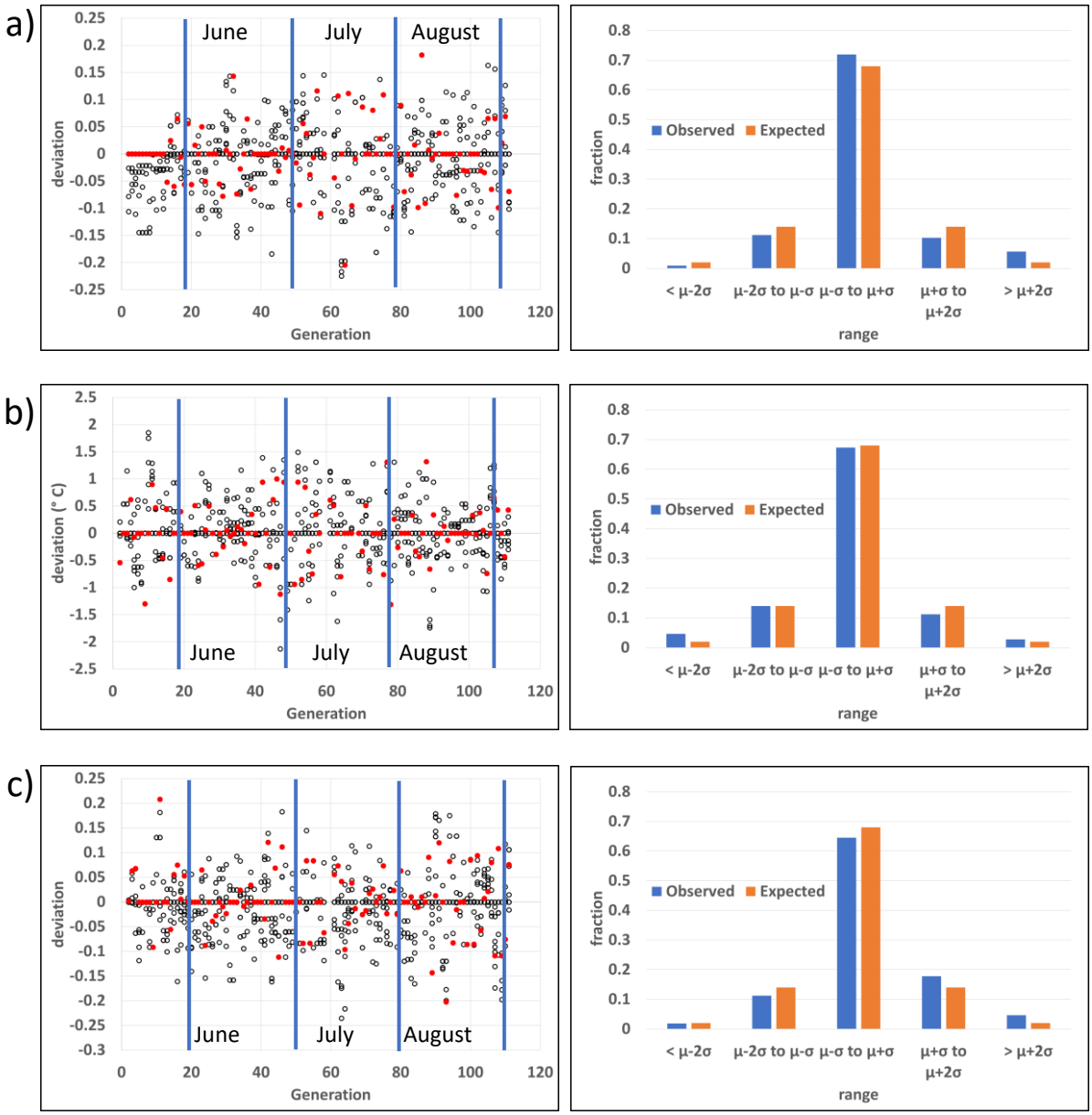


Figure 5 Change in parameter values of each member for each generation, with winning values in red (left), and distribution of winning changes with respect to all changes compared to what we expect from a Gaussian distribution (right) for a) AKMIN1, b) STGOFF, and c) Lm.

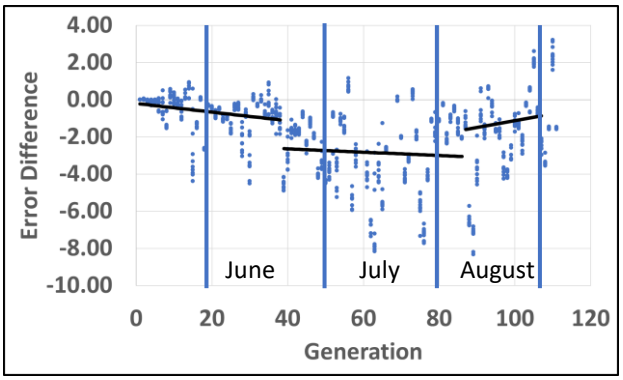


Figure 6 Difference in composite error score between the Alt member and each of the 10 members for each generation. A negative value indicates the ensemble member had a lower (better) score. The vertical blue lines indicate the first day of each month, with the series ending on August 31st. The trend lines are for the three periods as defined in the text.

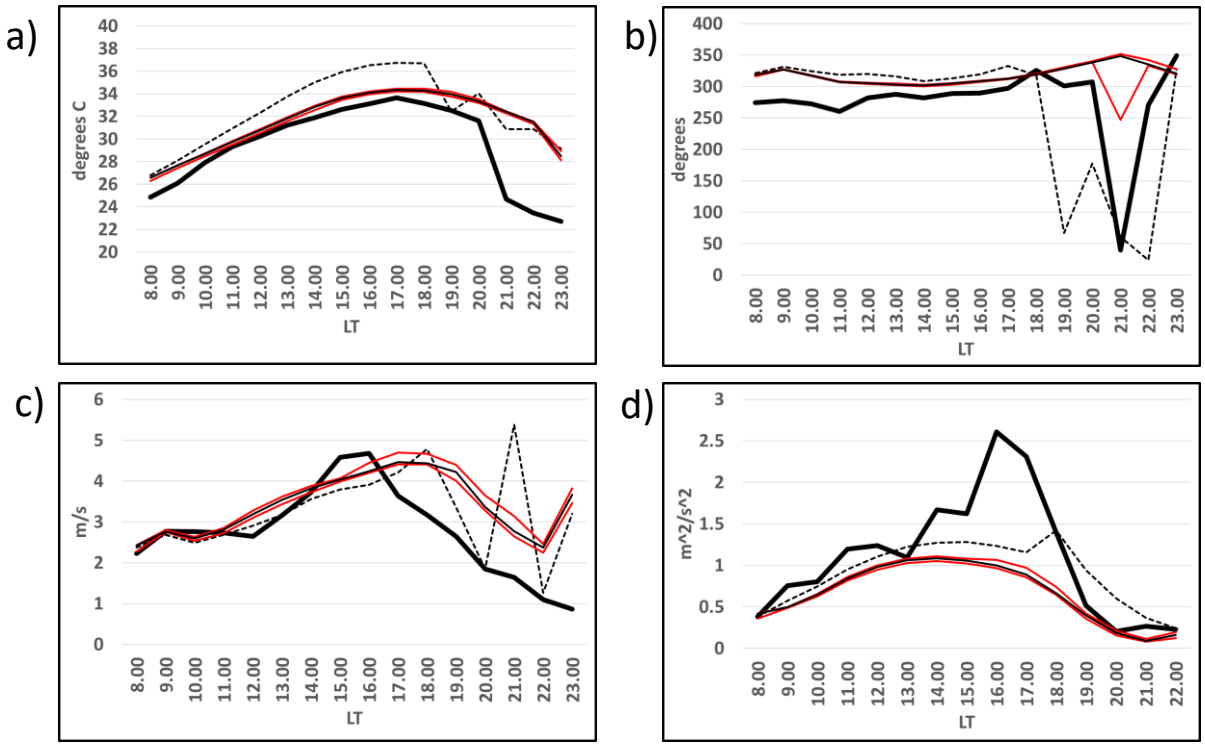


Figure 7 Time series for June 30th, for a) temperature, b) wind direction, c) wind speed at 18 m on the CL tower, and d) TKE at 61 m on the D tower. Data are for the best ensemble member (thin line), the Alt run (dashed line), and for observations (thick line), along with the maximum and minimum values from the ensemble (red lines).

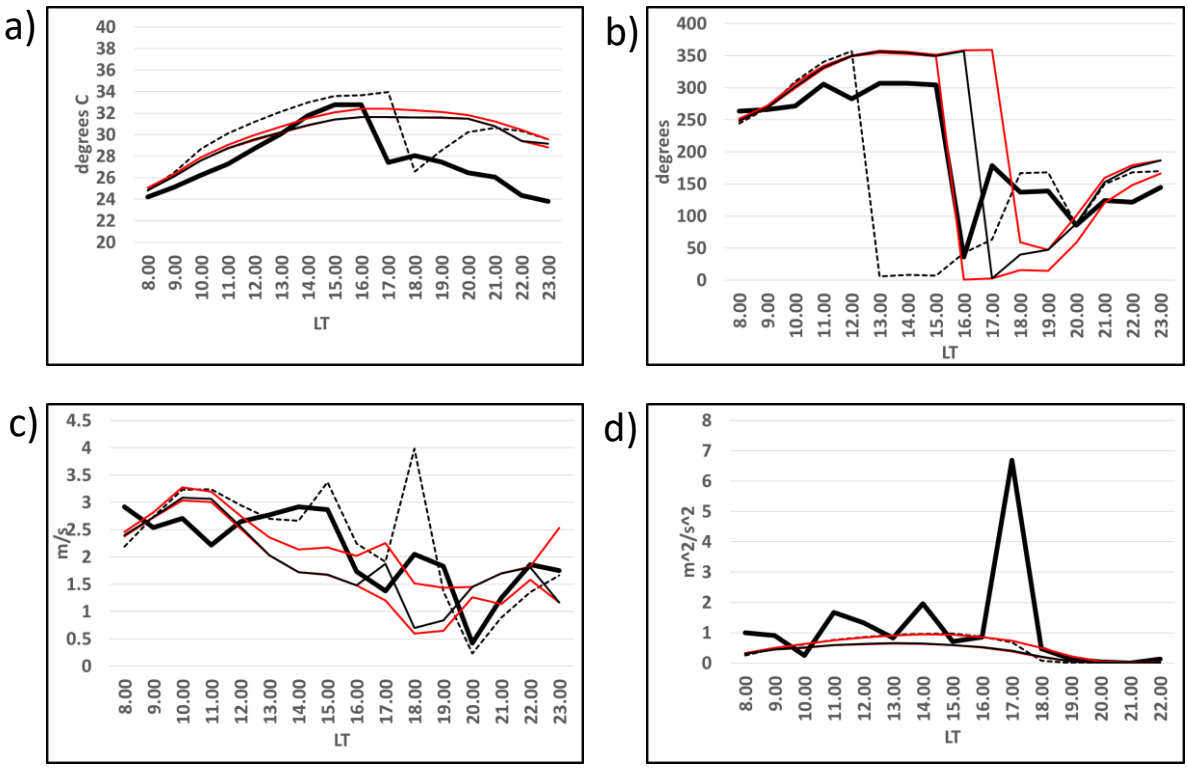


Figure 8 Time series for August 30th, for a) temperature, b) wind direction, c) wind speed at 18 m on the CL tower, and d) TKE at 61 m on the P tower. Data are for the best ensemble member (thin line), the Alt run (dashed line), and for observations (thick line), along with the maximum and minimum values from the ensemble (red lines).

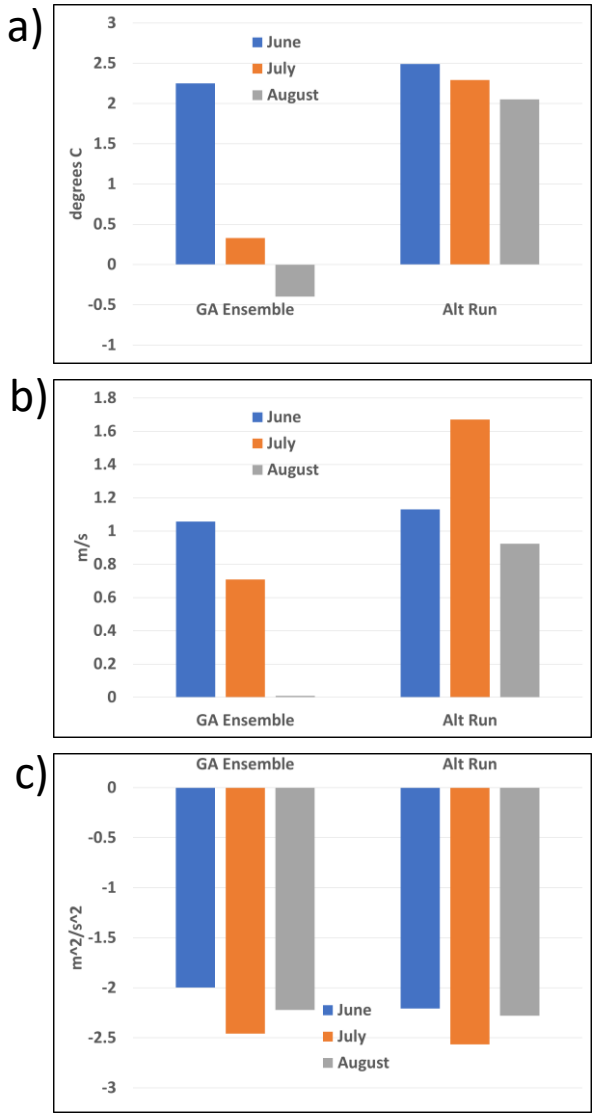


Figure 9 Ensemble mean of model bias for a) temperature, b) wind speed at the CL tower at 18 m, and c) TKE at the A tower at 61 m, as well as for the Alt runs, averaged over each month.

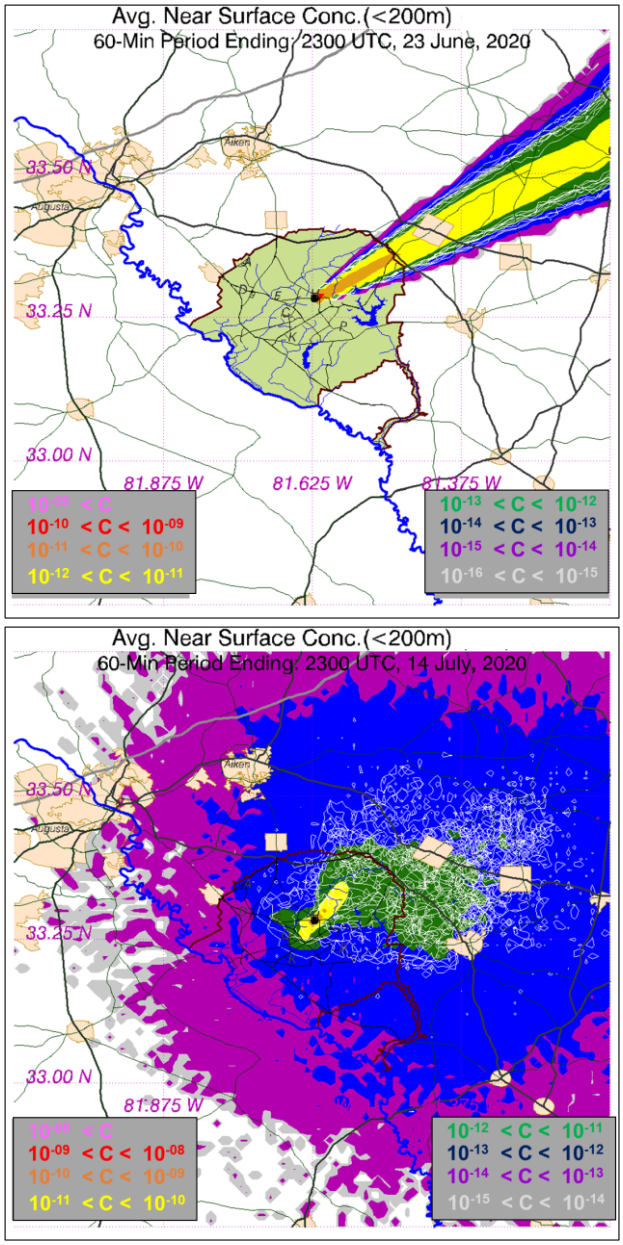


Figure 10 HYSPLIT ensemble-mean results (colors) for near-surface concentration (Ci/m^3) at 2300 UTC from the June 23rd (top) and 2300 UTC on July 14th (bottom). The white lines represent the contours indicated by green shading (>10^{-13} for June 23rd and >10^{-12} for July 14th) from each ensemble member. Note the difference in concentration scale for each figure.

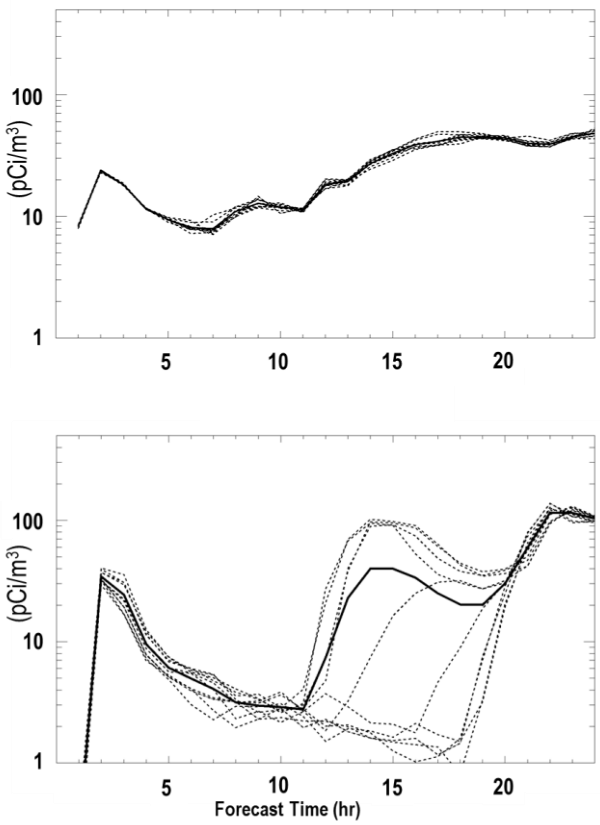


Figure 11 Maximum centerline concentration from the ensemble of HYSPLIT simulations along the SRS boundary for a) June 23rd, and b) July 14th. The dashed lines each represent one of the 10 ensemble members, while the bold solid line is the ensemble mean.

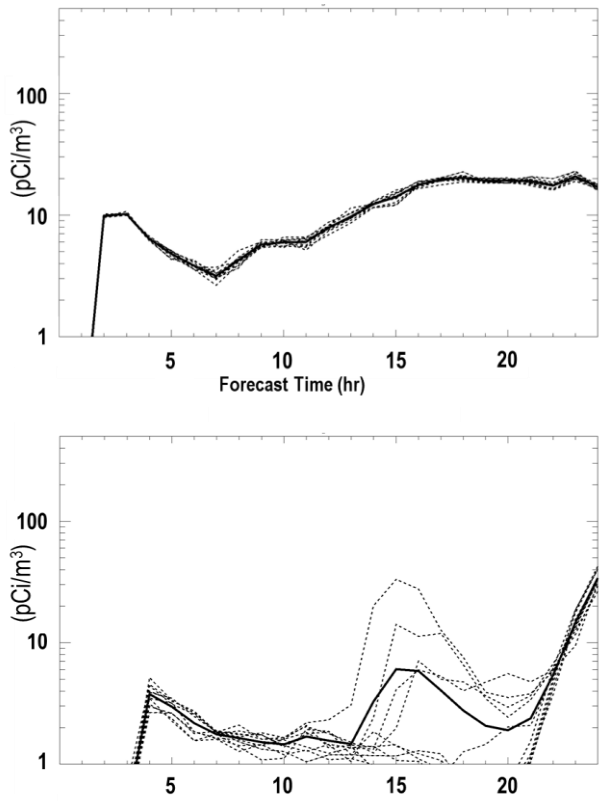


Figure 12 As in Fig. 11, but for data along a 30 km arc centered at SRS.

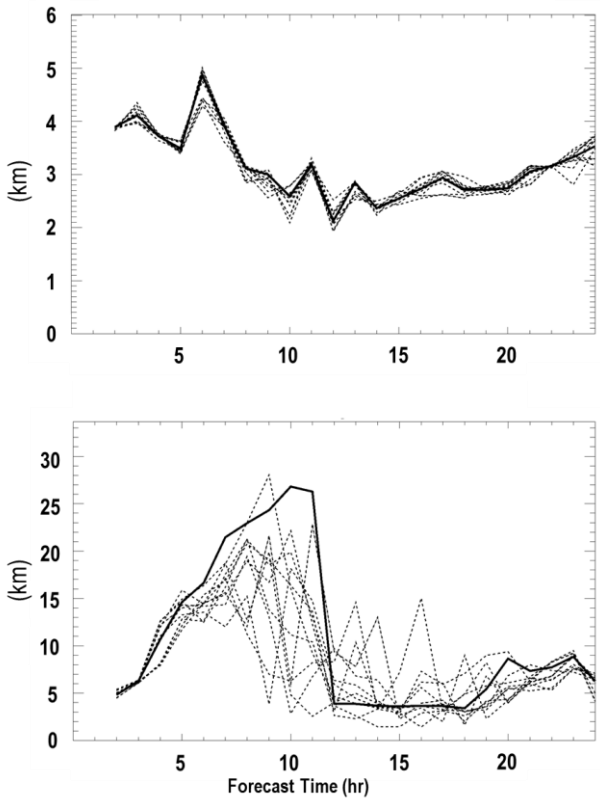


Figure 13 Plume half width from the ensemble of HYSPLIT simulations along the SRS boundary for a) June 23rd, and b) July 14th. The dashed lines each represent one of the 10 ensemble members, while the bold solid line is the ensemble mean. Note the different scaling for the two plots.

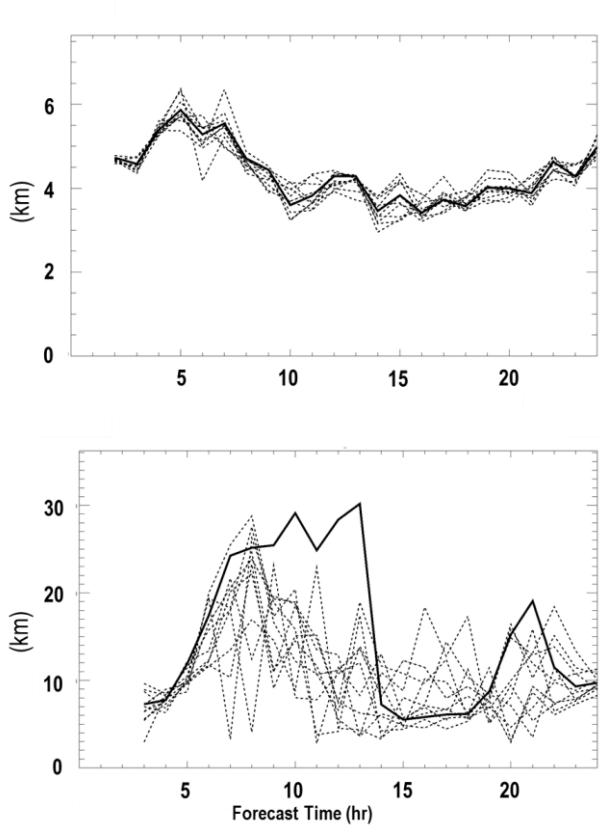


Figure 14 As in Fig. 13, but for data along a 30 km arc centered at SRS. Note the different scaling for the two plots.

Parameter #	Parameter	Permissible Range		Alt Value
		Minimum	Maximum	
1	AKMIN Grid 1	0.1	2.0	2.0
2	TNUDLAT (s)	900.0	1800.0	1350.0
3	TNUDCENT (s)	7200.0	14400.0	10800.0
4	SLMSTR	0.15	0.8	0.3
5	STGOFF (K)	-5.0	5.0	2.0
6	AKMIN Grid 2	0.1	2.0	1.0
7	TOPTENH	0.1	0.9	0.5
8	TOPTWVL	4.0	6.0	4.0
9	Lm	1.0	2.5	1.0

Table 1 Model parameter range (see Appendix A for a complete description), along with the Alt values. The latter are the unchanging parameter values used each day to get the Alt forecast, and are all within the recommended range of values as defined in the user technical manual.