

MAKING IFPS MORE CLIMATE AWARE
Jeffrey T. Davis, Weather Forecast Office, Tucson, AZ
September 10, 2002

Background

As part of the WFO Tucson's GFE Rapid Prototype Project (RPP) work, an initiative was started to investigate the use of climatology in smart tools, smart initialization, and text generation. The primary goal was to make the baseline IFPS more climate aware. In doing so, the usefulness of climate grids would have to be demonstrated to justify the extra costs and resources needed to implement the idea. The first step was to figure out how to make this information available in the IFPS. Fortunately, the GFE framework already had the potential to store static or persistent grids in the form of another IFPS database. As the GFE evolved through the RPP work, more flexible ways of customizing and accessing model grids through Python methods expanded the potential use of climate and GIS information in IFPS. Persistent grids of just about any type can be accessed by GFE smart tools and initialization algorithms. For example, 500 mb heights can be stored as a persistent grid and utilized in a pattern recognition algorithm. Gridded land use, soil moisture, and vegetation information can also be stored in an IFPS database and used in a surface layer parameterization scheme to help improve temperature initialization for high resolution forecasts.

Developing the GFE framework so that the National Weather Service (NWS) can make "smarter" model interpretation algorithms in the form of smart tools and smart initialization was one of the RPP goals. The development of these algorithms is the responsibility of the NWS and not the FSL software developers. Unfortunately, in an evolving system like the IFPS, priorities have to be established. Items like text generation, grid forecast verification, and web-based product presentation have been in the forefront. However, with the designation of a smart tool "Czar" and the on-line repository, improvements in grid initialization and editing should eventually take place. One could argue that this is not enough, but at least it is a "baby step" toward improving model first-guess fields and general editing tools.

Model Interpretation: Point-Based to High Resolution Grids

It was clear early on that most of the GFE grid-based model initialization algorithms were not capturing the basic spatial distributions of surface temperature, wind, and dew point. This was even more noticeable at higher grid resolutions. It was speculated that this might be attributed to the lack of good vertical model resolution in the PBL. The higher resolution grids looked more realistic, but without good first-guess model fields, the task of editing would remain difficult and time consuming. The lack of sufficient first-guess model fields created some frustration and forecasters were looking for more simplified ways to edit grids. The GFE was graphical and spatially-oriented, while the forecaster mind-set was more point-based. Some forecasters simply wanted to enter the maximum and minimum temperatures at several locations and let the "smart" algorithms do the rest. In principle, this is a great idea. We have surface analysis packages like ADAS and LAPS that take unevenly spaced point observations and interpolate them on a grid at a predefined resolution. Why not apply the same interpolation techniques that are used for point observations to point forecasts? Objective analysis techniques like the ADAS Bratseth method of successive correction that converges to an optimal interpolation based on a model (RUC) first-guess field and observational statistical errors could be utilized. Or just use a technique like the LAPS modified Barnes analysis. However, what if the first-guess model grid does not even capture the general spatial distribution of the field? Just add more points and

your interpolation scheme will work better. This sounds great, but more points for the forecaster to adjust defeats the purpose since it would not significantly decrease the workload.

Eventually it was argued that a point-based adjustment tool may not be the best approach. Instead, it was decided to focus on improving the MOS gridding technique since it was believed that even high resolution forecasts from mesoscale NWP models would require some post processing. This automated approach would allow for future enhancements in the area of optimally combining (blending) point forecasts using climatology, persistence, multi-model ensembles, and same model ensembles based on past performance. However, downscaling randomly spaced or equally spaced grid points to scales approaching landscape (generally below 3km) is not a trivial task.

Several valid techniques are available that can be used to weight grid points to station forecast points for the purpose of interpolation used in downscaling. MDL chose to develop a configurable station-to-gridpoint mapping scheme to grid MOS point guidance for the IFPS. This technique of subjectively assigning grid point weights works quite well for low grid resolutions (20km +) and flat terrain, but becomes a daunting task at higher resolutions over complex terrain. Thus, an alternative method of mapping point forecast guidance was necessary to support the higher resolution grids that were being used in the mountainous regions of the western United States. The idea of using climatology for mapping randomly spaced data is not new, but obtaining quality high resolution climate grids is not trivial.

High Resolution Climate Grids

Climatic mapping or gridding can be a shaky business, especially at high resolutions. Historically, climatic mapping often involved statistical and subjective interpretation by expert climatologists and cartographers using not only climate station data, but also general information of topography, land use, and land cover which were known to influence climate. This more subjective approach worked rather well for broad latitudinal and elevation-dominated trends, but generally failed to capture important details. Many mathematical techniques have been used for spatial interpolation of climatic point-based data, including B-splines, Kriging, distance weighting, and trend analysis. Less traditional methods utilizing 3D fractal interpolation, fuzzy logic, and neural networks have also been used. One of the more popular commercial climate mapping systems is PRISM (Parameter-elevation Regressions on Independent Slopes Model) which utilizes DEM data and a weighted regression technique to spatially distribute point climatology on a grid. The PRISM-generated climate grids can be purchased from www.climate-source.com and come in various GIS formats for ESRI's ARC/INFO and ArcView or GRASS down to a 2km resolution.

Because of the cost involved in obtaining PRISM climate grids and the need to formulate a proof-of-concept for the use in IFPS, WFO Tucson settled on using an overlapping-elevation multiple regression technique for gridding average monthly maximum and minimum temperatures. A similar approach was used for average monthly dew points and precipitation. A modeled wind climatology will also be developed using the MM5.

How to Store Static Grids in the GFE

The GFE has generally two methods in which a gridded field can be stored and accessed via the Python methods. One way is to treat the grid as a D2D or as another IFPS database. Since the D2D method was not available early on, the latter approach was used. By configuring a persistent database via the "localConfig.py" file, one can transfer grids with the GFE ifpAG

utility program. The following is a snippet from the localConfig.py file:

```
*****Define CLIMO Database*****
*****
#Define new CLIMO elements
MaxTClimo = ("MaxTClimo", SCALAR, "F", "Monthly Mean MaxT", 122, -30, 0)
MinTClimo = ("MinTClimo", SCALAR, "F", "Monthly Mean MinT", 122, -30, 0)
SVRClimo = ("SVRClimo", SCALAR, "percent", "Prob of Svr", 100, 0, 0)
FFClimo = ("FFClimo", SCALAR, "percent", "Prob of Flash Flood", 100, 0, 0)
PrecipClimo = ("PrecipClimo", SCALAR, "in", "Precip Amounts", 10, 0, 2)
#Define new time constraint for Climo Grids
staticTime = (0, 0, 0)
#Define database and parm list
CLIMO = ('CLIMO', GRID, "", YES, NO, 1, 0)
ClimoFields = ([[SVRClimo, FFClimo, PrecipClimo], staticTime),
([MaxTClimo], MaxTTC), ([MinTClimo], MinTTC)]
ADAS10 = ('ADAS10', GRID, "", YES, NO, 1, 30)
ADASParms = ([[Temp, Wind], TC1)]
ETA10 = ('ETA10', GRID, "", YES, NO, 2, 0)
ETA10Parms = ([[Temp, Wind], TC1)]
dbs = [(CLIMO, ClimoFields), (ADAS10, ADASParms), (ETA10, ETA10Parms)]
*****
```

Summary

Preliminary results using high resolution climate grids to spatially distribute point temperature and dew point forecasts look promising, and may provide an office with an alternative approach to the mapping MOS guidance. Future archival features and simple IO functions of the GFE DFC will open the door for even more point-based model interpretation methods like Dynamic MOS (DMOS) and ensemble blending that might utilize climate grids.

Unfortunately, many components in the IFPS are not plug-n-play. The same is true with any advanced features like storing high resolution climate grids or various GIS related information. However, the general framework has been put in place as part of the RPP development process. Most of the information you need for configuring the GFE for more advanced uses can be found in the FSL documentation.