All changes are inside spread.c.

Function: spr\_spread

Original code: for (i = 0; i < total\_pixels; i++)

{

if ((z[i] == 0) && (delta[i] > 0))

{

/\* new growth being placed into array \*/

temp2 += (float) slp[i];

z[i] = delta[i];

temp1++;

}

Parallel code: #pragma omp parallel for default(shared) reduction(+:temp1,temp2) schedule(dynamic, 2048)

for (i = 0; i < total\_pixels; i++)

{

if ((z[i] == 0) && (delta[i] > 0))

{

/\* new growth being placed into array \*/

temp2 += (float) slp[i];

z[i] = delta[i];

temp1++;

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 641.74

Function: spr\_GetDiffusionValue

Original code: rows\_sq = igrid\_GetNumRows () \* igrid\_GetNumRows ();

cols\_sq = igrid\_GetNumCols () \* igrid\_GetNumCols ();

Parallel code: #pragma omp parallel num\_threads(2)

{

#pragma omp sections

{

#pragma omp section

{

rows\_sq = igrid\_GetNumRows () \* igrid\_GetNumRows ();

}

#pragma omp section

{

cols\_sq = igrid\_GetNumCols () \* igrid\_GetNumCols ();

}

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 711.76

Function: spr\_urbanize

Original code: nrows = igrid\_GetNumRows ();

ncols = igrid\_GetNumCols ();

Parallel code: #pragma omp parallel num\_threads(2)

{

#pragma omp sections

{

#pragma omp section

{

nrows = igrid\_GetNumRows ();

}

#pragma omp section

{

ncols = igrid\_GetNumCols ();

}

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 676.14

Function: spr\_get\_neighbor

Original code: nrows = igrid\_GetNumRows ();

ncols = igrid\_GetNumCols ();

Parallel code: #pragma omp parallel num\_threads(2)

{

#pragma omp sections

{

#pragma omp section

{

nrows = igrid\_GetNumRows ();

}

#pragma omp section

{

ncols = igrid\_GetNumCols ();

}

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 656.13

Function: spr\_phase4

Original code: nrows = igrid\_GetNumRows ();

ncols = igrid\_GetNumCols ();

Parallel code: #pragma omp parallel num\_threads(2)

{

#pragma omp sections

{

#pragma omp section

{

nrows = igrid\_GetNumRows ();

}

#pragma omp section

{

ncols = igrid\_GetNumCols ();

}

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 718.46

Function: spr\_phase5

Original code: nrows = igrid\_GetNumRows ();

ncols = igrid\_GetNumCols ();

Parallel code: #pragma omp parallel num\_threads(2)

{

#pragma omp sections

{

#pragma omp section

{

nrows = igrid\_GetNumRows ();

}

#pragma omp section

{

ncols = igrid\_GetNumCols ();

}

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 679.96

Function: spr\_spiral

Original code: nrows = igrid\_GetNumRows ();

ncols = igrid\_GetNumCols ();

Parallel code: #pragma omp parallel num\_threads(2)

{

#pragma omp sections

{

#pragma omp section

{

nrows = igrid\_GetNumRows ();

}

#pragma omp section

{

ncols = igrid\_GetNumCols ();

}

}

}

Original cost of function spr\_spread: 692.87

The cost of parallel version: 833.71