SERT models summary

1. _d_
   - 1.1 _eco_
     - 1.1.1 _p_
     - 1.1.2 _s_
     - 1.1.3 _t_
   - 1.2 _h_
   - 1.3 _p_
   - 1.4 _s_
   - 1.5 _sup
2. _ftg_
3. _fm_
4. _mt_
5. _sd_

Total models = 5*5*3 = 75

Nomenclature:

Built Form		
_d_	Detached	
ftg	Flat (top/ground)	
_fm_	Flat (middle)	
_mt_	Mid-terraced	
_sd_	Semi-detached	
Differences in		
*.cnn *.obs *.shd		
*.geo (different thbr widths)		

SAPP models overview

Introduction

There are 7 design parameters with the following permutations:

#	Key	Parameter	Files changed	Permutations

1	X	Exposure	*.geo	5
2	I	Insulation	constr.db, *.con, *.tmc	5
3	A	Air tightness	*.opr	3
4	C	Thermal capacity	constr.db, *.con, *.tmc	2
5	S	Solar ingress	*.geo, *.shd	5
6	O	Occupancy density	*.opr	5
7	L	Living area fraction	*.geo	5

1. There are different cfg files for each permutations. Support files (i.e. all files that are not cfg files) are of two types: primitive and generated. Only the primitive files are present in the subversion archive.
2. Generated files are mostly created using prerun_tasks.sh

What the scripts do?

Figure 1 shows schematically how the process works.
cfg_maker.sh creates the cfg files alongwith relevant references to various support files. All cfg files are created from the primitive template `configuration.file`
It can be called with the argument `automatic` in which case all models are made automatically and `prerun_tasks.sh` is also called with the argument `automatic`. The script `check.sh` is also called when `prerun_tasks.sh` returns (currently disabled).

prerun_tasks.sh creates the following support files:

1. All *.con files
2. Runs ish and creates binary shading databases
3. All *.opr files to create the A and O permutations. Two primitive operations templates are used for this purpose.
4. *.geo files to create S and L permutations
5. For any `*.clm.a` ascii climate files in the `../dbs` folder the script converts these to binary clm files deleting the ascii versions. Finally it calls `clm_writer.awk` to write all regression relevant independent climate parameters to `clm.summary.data.a`

This script can also be called with the argument `automatic` in

which case all the above actions are performed automatically.
The script can also be called with the arguments

`model` – automatically runs model specific prerun_tasks (1-4 above) only

`climate` – automatically runs climate specific prerun_tasks (5 above) only

`hpc` – generates scripts that will run simulations on a HPC

check.sh runs simulations. The user can choose to run all the models or a subset by providing ranges for XIACSOL. The user can also choose to run in test mode where the models to be run are hard coded in the script. The range can be modified by editing this script. The script accepts no command line arguments.
After simulating each model it calls `xtract.awk` to extract results to `results.file` which is processed by `mk_RRDB.pl` to create raw results database `RRDB.csv` which holds model no., climate name and monthly heating and cooling loads.

Cleanup.sh interactive or automatic cleanup of all generated files.

How to change Design Parameter values?

All design parameters can be changed wither by changing some value in the primitive support files or prerun_tasks.sh

To change	Edit	Comments
X	*.geo, *.cnn	
I	I?C?constrdb	Change thickness of insulation and run prerun_tasks.sh to rebuild *.con and *.tmc files
A	prerun_tasks.sh*	Edit ach in relevant section (variable A? and AT? for integral and fractional part respectively).

C	I?C?constrdb	Change the material properties in the database and rebuild all *.con and *.tmc files
S	prerun_tasks.sh*	Edit bottom coordinates for windows (variables Sol? And SolT? For integral and fractional part respectively)
O	prerun_tasks.sh*	Edit occupancy type casual gains (variable name is Occ*)
L	prerun_tasks.sh*	Edit coordinates of corner points common to both zones (variable name is Geo?)

*Note that prerun_tasks changes copies templates to support files and then modifies copied support files to make various permutations. Any changes in prerun_tasks should also be reflected in the templates. For example 0.65ach is the value in operations templates, prerun_tasks copies the templates for A1 and then changes 0.65 to 1.00 and 1.50 to make A2 and A3. To change A1 permutation 0.65 should be edited in the template and then for A2 and A3 the 0.65 substitution with 1.00 and 1.50 should be changed in prerun_tasks.

Adding more zones

More zones can be added. They should be names *liv where * is any string. ‘liv’ is the string that is used in the cfg_maker and prerun_tasks scripts to modify file name references and hence should be kept at the end of zone name.
Make sure that static/primitve files for all new zones exist. Non-primitive files can be erased by running the cleanup script. Surfaces associated with the new zones should also be added in all the primitive connections files and zone file references should

be added in cfg template. Also find occurances of ‘nliv’ in *.sh and add the new zone if appropriate (the second zone is called nliv)

Model creation, simulation and creation of databases

1. Make sure SAPP ascii climates are present in the dbs folder. The climate file names are uk_SAPP_*.clm.a
2. Run cfg_maker.sh (manually and [y] to all questions)
3. Run prerun_tasks.sh (interactively and [y] to all questions, note that prerun_tasks need not be run if cfg_maker is asked to run automatically – question is asked when cfg_maker runs).
4. chmod a+x or 744 *.sh to make sure that HPC scripts are executable.
5. cd to folder above SAPP folder and make folders for each climate. Now go into each folder and run the hpc script (make sure that binary climate files are present in dbs folder before running simulations this should be made by prerun_tasks.sh either interactively or by the command line argument climate). These two are included in the scratch script that needs to be edited manually before running. Note that the HPC script calls check.sh to simulate models. check.sh also runs xtract.awk to extract the results to results.file and calls perl script to create RRDB.uk_SAPP_[climatename].csv
6. Copy all RRDB.csv files into the SAPP/cfg folder and delete all temperary folders SAPP_[climatename]

cp ../../SAPP_*/cfg/*.csv . [from SAPP./cfg folder]
rm -rf ../../SAPP_*

7. Run g77 MLR.F to compile the MLR program, this generates an executable usually called a.out
8. Run a.out but first making sure that clm.summary.data.a is present (made by prerun_tasks.sh) and all RRDB.csv files are present in the cfg folder. Also make sure that there is no file by the name FRDB.csv before running a.out

Adding more climates

Add the ascii climate files to ../dbs folder following the naming convention uk_SAPP_[climatename].clm.a
Run prerun_tasks.sh with command line argument climate (this makes climate binary files)

Changing number of XIACSOL

Make changes in:

- cfg_maker.sh
- prerun_tasks.sh
- MLR.F
- check.hpc.sh

Climate files

Climate files are obtained from post processing files generated by the UKCIP09 Weather Generator (WG) software.
The WG is available at:
http://ukclimateprojections-ui.defra.gov.uk/ui/admin/login.php
The WG makes 100 climate files each of which is an equi-probable set of future scenarios for 30 years. The base line set consists of 30 years of historic climate data which is also simulated. Climate files are built by the WG as follows:
Start page > new request > data source > WG simulatins > standard WG variables > low > 2020s > choose grid cells > generate files

Generated files are as follows:
R_????_cntr_hly.csv (100 files control or historic hourly data)
R_????_scen_hly.csv (100 files scenario or predicted hourly data)
R_????_cntr_dly.csv (100 files control or historic daily data)
R_????_scen_dly.csv (100 files scenario or predicted daily data)
Daily files are not required and may be deleted. Present work is based on historic data but may be extended to future predicted data as well.

These files can be parsed by the script **clm_builder.bash** which calls **clm_xtractor.awk**. These scripts convert WG output files to ESP-r climate files. The script only converts one year at a time so has to be manually edited and run 30 times – to convert 30 years weather data.

Prototype Details

X – Exposure

Surface	1	2	3	4	5
liv_w_e		similar	similar	similar	similar
liv_roof					similar
liv_floor				similar	similar
nliv_w_s		similar	similar	similar	similar
nliv_w_w			similar	similar	
nliv_roof					similar
nliv_floor				similar	similar

(surfaces not marked similar are external)

I -Insulation levels (U values) [BA]

wall	2	0.6	0.31	0.31	0.13
roof	2	0.35	0.16	0.14	0.11
floor	0.88	0.46	0.25	0.22	0.13
glazing	0.56	3.4	2.1	1.7	1.1
door	3	3	2	1.7	1

A – Airtightness (ach) [BA]
0.1, 0.25, 0.5, 0.75, 1.0

C – Location of thermal capacity
Inside, middle

S – Solar ingress (window sizes) [BA]
15%, 22.5%, 32% of TFA

O – Occupancy levels (as per SAP i.e. function of TFA) [BA]
60, 80, 100, 150, 200m2

L – Living area fraction [BA]
0.1, 0.25, 0.5

ESP-r file structure

	cfg	cnn	geo	con tmc	shd shda	opr
X	5	5	5			
I	5			5		
A	5					5
C	2			2		
S	3		3		3	
O	5					5

L	5		5			
	18750	5	75x2*	10x2*	3x2*	25x2*

* for each zone