Reference	Name in PySulfSat	Melt composition?	T-sens?	P-sens?	H$_2$O-sens?	*Fe^{3+} sensitive?*	Sulfide/Sulfate comp?	Cali dataset available?
SCAS models								
Chowdhury & Dasgupta (2019)	"calculate_CD2019_SCAS"	✓	✓	✗	✓	✗	✗	✓
Zajacz & Tsay (2019)	"calculate_ZT2022_SCAS"	✓	✓	✗	✓	✗	✗	✓
Masotta & Keppler (2015)	"calculate_MK2015_SCAS"	✓	✓	✗	✓	✗	✗	✓
SCSS models								
Li and Zhang (2022)	"calculate_LiZhang2022_SCSS"	✓	✓	✓	✓	✓	✓	✓
Blanchard et al. (2021)	"calculate_B2021_SCSS"	✓	✓	✓	✓	✗	✓	✓
O'Neill (2021)	"calculate_O2021_SCSS"	✓	✓	✓	✗	✓	✓	
O'Neill and Mavrogenes (2022)[*1]	"calculate_OM2022_SCSS"	✓	✓	✓	✗	✓	✓	✓
Liu et al. (2021)	"calculate_Liu2021_SCSS"	✗	✓	✓	✓	✗	✓	✓
Smythe et al. (2017)	"calculate_S2017_SCSS"	✓	✓	✓	✓	✓	✓	✓
Fortin et al. (2015)	"calculate_F2015_SCSS"	✓	✓	✓	✓	✗	✗	✓
Sulfide composition models								
O'Neill (2021)	"Calc_ONeill"	✓	✗	✗	✗	✓		
Smythe et al. (2017) using Kiseeva et al. (2015)	"Calc_Smythe"	✓	✓	✗	✗	✓		

Calculating Proportion of S^{6+} using empirical approaches

Reference	Name in PySulfSat	Input parameters
Jugo et al. (2010)	"calculate_S6St_Jugo2010_eq10"	ΔQFM
Nash et al. (2019)	"calculate_S6St_Nash2019"	T, Fe^{3+}/Fe$_T$
O'Neill and Mavrogenes (2022)	"calculate_OM2022_S6St"	Melt comp, T, log(fo$_2$) **or** Fe3/Fe$_T$

Correcting SCSS^{2-} and SCAS^{6+} calculations for S$_T$

Name in PySulfSat	Input arguments
"calculate_SCSS_Total"	SCSS^{2-}, S^{6+}/S$_T$
"Calculate_SCAS_Total"	SCAS^{6+}, S^{2-}/S$_T$
"Calculate_S_Total_SCSS_SCAS"	SCSS^{2-}, SCAS^{6+}, S^{6+}/S$_T$, or model ('Nash', 'Jugo', 'OM2022', 'Kleinsasser')

Other functions

"crystallize_S_incomp"	Calculates S left in the melt for a given F$_{melt}$ (assuming S is entirely incompatible
"calculate_mass_frac_sulf"	Calculates mass fraction of sulfide removed for a model of changes in SCSS with fractional crystallization
"convert_d34_to_3432S", "convert_3432S_to_d34"	Converts δ^{34}S to ^{34}S/^{32}S and vice versa
"Lee_Wieser_sulfide_melting"	Modelling of S and chalcophile element behaviour during mantle melting.
For Monte Carlo simulations	
'add_noise_2_dataframes'	Generate duplicated rows in df1 based on errors present in df2
'add_noise_series', 'duplicate_dataframe'	Used to simulate uncertainty in specific variables
'av_noise_samples_series'	Average outputs from Monte Carlo simulations per sample