

任意の材料組合せ材の熱貫流率算定

AutoACJ プログラムには、任意の材料組合せ材の熱貫流率が、計算できる機能が含まれている。

CWallCompDlg クラス

任意の材料組合せのデータ群を管理しているクラス。

管理しているデータの種類

- 組合せ材の略号データ
- 組合せ材の概要説明データ
- 組合せ材の熱貫流率データ

この3種類のデータは、拡張できる配列(m_Array_WallComp)に保存される

この配列を外部ファイルに落とし、AutoACJ は略号を自動検索し、負荷計算に使用している。

事前に登録されている組合せ材は、略号で呼び出され、その都度、熱貫流率を計算する必要はない。

登録されていないものが必要なときは、負荷計算をする前に、材料を選定し（以下の機能を使い）登録すれば、ほぼ任意の組合せ材を使って計算ができる。

この貫流率計算と登録は、以下に説明する CThermalTransDlg クラスが行っている。

CThermalTransDlg クラス

任意の材料を選定し、熱貫流率などを計算する。

材料選定は、エクセル調の入力ダイアログツールでおこなう。

選定材料のデータは文字列データとして扱っている。

- 材料名 CString m_compo 文字列
- 熱伝導率 CString m_kvalue kcal/m・h・℃
- 厚み CString m_thick m
- 容積比熱 CString m_Cvalue kcal/m³・℃ 応答係数計算のため追加した

この4種類のデータは、拡張できる配列(m_ArrayThermalTransData)に保存される。

選定出来る材料は、

- 金属-3種類
- 非金属-20種類

- その他-8種類が
が、プログラム内部に登録されている。(この登録は変更できるように改造が望ましい)
- また、貫流率計算に必要な、
- 外表面熱伝達率 (入射抵抗)
 - 内表面熱伝達率 (輻射抵抗)
- も部位カテゴリ別に選定できるようにしてある。

カテゴリ	外表面熱伝達率(α_0) Kcal/h · m ² · °C	内表面熱伝達率(α_i) Kcal/h · m ² · °C
垂直外壁	15.0	7.12
屋根	20.0	5.27

登録されている材料データ

カテゴリー	名称	熱伝導率(λ) Kcal/m \cdot h \cdot °C	容積比熱 Kcal/m 2 \cdot °C
金属	アルミ板	180.0	567
	銅版	320.0	819
	鉄板	47.0	821
	ステンレス鋼	13.0	766
非金属	ブリック (シンガポール基準)	0.992	332
	軽量コンクリート 1種	0.7	447
	普通コンクリート	1.41	481
	ALC 板	0.11	156
	モルタル (セメント)	1.1	551
	モルタル (パーライト)	0.31	380
	タイル	1.1	624
	大理石	2.4	561
	グラナイト	2.5	562
	合板・化粧板	0.16	266
	木材 (中量材)	0.15	155
	木材 (重量材)	0.16	186
	板ガラス	0.68	483
	スレート	1.2	435
	プラスタボード (PB)	0.19	204
	フレキシブルボード	0.6	311
	パーティクルボード (A5908)	0.15	171
	木毛セメント板 (普通品)	0.16	147
	たたみ	0.130	69
	カーペット	0.069	76
	PVC 床タイル	0.163	350
その他	空気層 (密閉中空層)	5.0	0
	空気層 (非密閉中空層)	10.0	0
	グラスウール保温材 (32K)	0.04	6.4
	ロックウール保温板 1, 2号	0.03	20
	岩綿吸音板-天井ボード	0.055	60
	スタイロフォーム	0.032	12
	硬質ウレタンフォーム	0.024	11.3

	ポリエチレン保温板(1種)	0.034	8.4
	防水アスファルトルーフィング	1.054	255

これらのデータは、通産省昭和55年2月29日 from ECCJ 及び
 日本建築学会：建築学便覧、小原俊平：建築の熱設計、渡辺 要：建築計画言論 II、
 日本建築学会編：建築資料集成 2、渡辺荘児・中島康孝他3名：蓄熱材料における土壌
 の熱的研究(3) などより出典している。
 資料によっては値が異なるが、近似のデータを代用して計算する事をお勧めする。

参考

熱伝導率(λ) [kcal/hm°C] Thermal Conductivity

熱抵抗(R) [m²h°C/kcal] Thermal Resistance

部材の厚み[m] / 熱伝達率[kcal/hm°C] = 熱抵抗[m²h°C/kcal]

複合部材の熱抵抗は上記の式で計算する(ΣR)

熱伝達率・熱伝達係数 ($\alpha_0 \cdot \alpha_i$) [kcal/h m²°C] film coefficient

空間から物質の表面に熱が移動する量

複合部材の総合熱抵抗は、 $TR = \frac{1.0}{\alpha_0} + \sum R + \frac{1.0}{C} + \frac{1.0}{\alpha_i}$

熱貫流率(K) [kcal/h m²°C] $U = \frac{1.0}{TR}$

温度の高い空間から複合部材を介して温度の低い空間に熱が移動する量

記号は、国際的には、Uの記号が使われている、最近国土交通省でもUとしている。

熱貫流率の計算

AutoACJ では、熱貫流率の計算の方法は、ツールバーの  ボタンを押す事で始めますが、次のように2通りあります。

1. あらかじめ部材の組合せを決め、熱貫流率をマニュアル計算し AutoACJ の Data フォルダ(Component Data. cmp)に書き込んでおく。この場合、応答係数は計算されないため、実効温度差は自動で計算できない。
2. 入力ダイアログの右下にある **熱貫流詳細計算** ボタンを押すと、新たなダイアログが現れ、任意の材料の組合せによる計算が出来ます。

応答係数の算出

従来の負荷計算は、あらかじめ設定された応答係数を使用して算出していました。しかしながら、応答係数は、部材の組合せで相当違う為、正式な応答係数を算出する事にしました。

AutoACJ プログラムを大きく変更せず、この機能を盛り込むには、上記の 2. 熱貫流率詳細計算の部分に追加する事としました。

1. **熱貫流率詳細計算** ボタンを押すと `CWallCompDlg::OnClickBThermalTrans()` が呼ばれ新たなダイアログが現れ、部材を選定する中で、その部材の組合せによる、24 時間分の応答係数を計算する事にしました。
2. `CW_Material` クラスを新設し、そのメンバ変数に、上記で選定された任意の部材のデータを格納します。このメンバクラスは、
 - `double` `mB`; 部材の厚み (m)
 - `double` `mH`; 部材の熱伝導率 (kcal/hm°C)
 - `double` `mC`; 部材の容積比熱 (kcal/m³°C)
4. `CDyResponse` クラスを新設し、応答係数を計算します。計算されたデータのうち負荷計算で必要となるデータを、転送する新たな `CRes_Factor` クラスも新設しました。`CRes_Factor` クラスでまとめられるデータは、以下の通りです。
 - `CArray <CW_Material, CW_Material> m_Array_WM`; 部材の組合せ詳細
 - `CString m_szName`; 組合せの略号
 - `double m_KV`; 熱貫流率
 - `double m_Factor[24]`; 24時間分の応答係数

このクラスは、`ThermalTransData.h` ファイルの先頭に定義しましたので、このメンバ変数を使うときは、`#include "ThermalTransData.h"` をコーディングする必要があります。

応答計算の為、`ThermalTransDlg` クラスに、以下の関数を追加しました。

```
CThermalTransDlg::GetWMaterials()
```

貫流率計算のときの部材のデータを取り込む

```
CThermalTransDlg::Start_ResponseF()
```

応答係数の計算の本体である、`CDyResponse` クラスを呼び出します。

熱貫流詳細計算

ボタンを押したときに、表示されるグリッドの「OK」ボタンを押したとき、計算の結果は、CRes_Factor クラスメンバに代入され返されます。ただし、略号は、この時点では空白で返されます。

熱貫流率詳細計算が終了すると、CWallCompDlg::OnClickBThermalTrans() 関数に制御が戻ります。

そして、熱貫流率データ入力用のグリッドに、計算された貫流率の値が書き込まれます。

```
Update_ThermalTrans(); // 還流率計算をフィードバック
```

部材の組合せ略号の取得は、CWallCompDlg::GetWallCompData() 関数で取り込み、略号が重複しないように、CWallCompDlg::Check_Code() でチェックされます。

その後「壁等材質登録」のダイアログで、「OK」ボタンを押すと、CWallCompDlg クラスのメンバ変数 “m_rpf” に CRes_Factor クラスで定義されている内容が全て書き込まれ返されます。

負荷計算のときに必要となるデータは、略号と応答係数の組合せデータです。

これらのデータは、従来から使っている、“Component Data. dat” という外部データファイルに書き込むようにします。

外部ファイル作成部分の CWallCompDlg::SaveCompData() 改造

組合せ部材のデータは、CWallCompDlg クラスで定義している m_Array_WallComp に書き込まれています。データは文字列データとして 3 種類の情報が扱われています。

この部分に、**応答係数データを追加**すれば、負荷計算にフィードバックできるはずですが。その準備として、m_Array_WallComp の構成クラスに、応答係数を「24」配列として以下のように追加します。

- CString m_number // コード略号;
- CString m_detail // 材料説明
- CString m_uvalue // 熱貫流率
- double m_Factor[24] // 応答係数

更に、外部ファイルに書き込む関数の、CWallCompDlg::SaveCompData() で以下の修正が

必要になります。

```
for(int i=0; i<nIndex; i++) {
    szNumber = m_Array_WallComp[i].m_number; // 略号
    szDetail = m_Array_WallComp[i].m_detail; // 部材組合せの説明
    szUvalue = m_Array_WallComp[i].m_uvalue; // 熱貫流率
    szData.Format("%-5.5s, %-30.30s, %-9.9s, ", szNumber, szDetail, szUvalue);
    stdfile.WriteString(szData);
    // 応答係数データ書き込み(144 文字)
    for(int j=0; 24>j; j++) {
        factor = m_Array_WallComp[i].m_Factor[j]; // 応答係数
        szData.Format("%-2.3f, ", factor);
        stdfile.WriteString(szData);
    }
    stdfile.WriteString("\n"); // 改行
}
```

これで、外部ファイル”Component Data. dat”に、応答係数データが含まれた一連のデータファイルが作成されます。

OWJ1	, ALC75+空隙+GW25+空隙+PB12	, 0.5607	, 0.006, 0.124, 0.218, 0.183,
略号	部材の説明	熱貫流率	応答係数 (24 時間分連続)

外部ファイル作成のプログラムでの動作フロー

ツールバーの  ボタンで、壁材などの登録開始

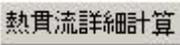
↓

ReadCompData() 外部登録ファイルを読み込み、新たな m_Array_WallComp 変数作成

↓

GetWallCompData() 上の m_Array_WallComp よりグリッド表示

↓

新たな部材登録のグリッドを表示、右下の  ボタン起動

↓

OnClickedBThermalTrans() 熱貫流率詳細計算開始

↓

ThermalTransDlg クラス 詳細計算グリッド表示

詳細計算ならびに応答係数算出 m_rpf 変数に代入して返される



Update_ThermalTrans() 部材登録グリッドに、貫流率表示、略号の入力要求



OnUpdateWallCompDataInput() 略号入力



OnOK()



Check_Code() 同じ略号を防止



SetWallCompData() 部材登録グリッドのカーソル制御



RecordData() m_Array_WallComp の追加が行われる



SaveCompData()

ファイルセーブのダイアログ表示して、外部データファイルの書き込み



外部データファイル作成終了

応答係数を負荷計算に利用する

壁・天井・床の負荷計算は、

CAutoACDoc::ACLoadCalcu()クラスで行っている。



ACLoadCalcu_Sub()



WallLoadCalcu()



WallLoadCalcu()



WallLoadCalcu_sub()



etd = CreateEtdData()

```

↓
// ***** 登録済み部材リスト部材名より応答係数取得
int nCount = Get_MNumber(type);
for(int j=0; 24>j; j++){
    if(nCount != -1){ // 登録部材が見つかった場合
        Rt[j] = m_ArrayWallComp.GetAt(nCount).m_Factor[j]; // 応答係数
    }
    else{ // 登録部材がない時、応答係数は=0 とする
        Rt[j] = 0.0;
    }
}
↓
for( int dt=0; 24>dt; dt++ ) { // 24時間に変更要
    int j_tm = ( time - dt -1 );
    if( j_tm < 0 )
        j_tm += 24; //1時の手前は時とする
    double aa = 0.0;
    aa = Rt[dt]*tn[j_tm]; // 応答係数(Yj) x (tn-j)を計算
    etd += aa; // Σ[(Yj) x (tn-j)]
}

```

簡易応答係数法は、12 時間の応答しか扱っていなかった。今回は、密度の大きい壁などの対応とするため、24 時間応答係数とする。

部材データ外部ファイル(WallCompData.cmp)で登録されていない場合、応答係数は、“0”とした。更に、登録されていない略号が使われた場合、負荷計算完了した時点で、警告をだし、修正を促すようにしてある。

ACLoadCalcu()関数で、プログレスバー表示終了の後に、メッセージボックスをコードして、行っている。

従来の応答係数は、SunApp.dll に書かれていたが、使い勝手が悪いので AutoACDoc クラス内に、関数を新設して対応している。

SunApp.dll の既存コードは、バンドルしている SunJ で使われているので、そのままにしておく。