Modelling Air Flows and Buildings with NMF and IDA

Axel Bring **Per Sahlin**

Building Engineering Services
Royal Institute of Technology

ABSTRACT

New object oriented simulation environments offer dramatically improved possibilities for simulation of coupled systems. In contrast to traditional building simulation, where separate, stand-alone tools are used for each simulation task; the new environments will offer a unifiedf ramework for all simulation problems. For design tool users this leads to two major advantages. (1) model coupling and comparison of results between different simulation applications will be practical, and (2) usage and input data will be standardised across applications. Thereby tool usability will increase considerably. Furthermore, man-time spent on model development and maintenance is expected to decrease, making it feasible to tailor appropriate models to each task and thus improve the quality of simulation results. In this paper, we report our experiences implementing a set of models for multi-zone air-exchange ME) in the general simulation environment IDA. Although MAE analysis is yet far from a standard industrial practice, several tailored tools have been developed. For comparison between general programs and specialised, the MAE models are extremely challenging, since the model structure can be utilised to an unusual degree in the specialised tools. In spite of this, the results obtained are encouraging in terms of processing time as well as numerical reliability (robustness). The paper presents a selection of the models, including the full Neutral Model Format (NMF) code, a discussion of numerical methods, and some general conclusions with bearing on simulation tool development.

1. INTRODUCTION

Studies of inter-zonal air flows in multi zone buildings have attracted interest for considerable time. Several tools for handling of these problems have emerged during the last decade, e.g., AIRNET (Walton 1989), Movecomp (Bring and Herrlin 1991), and COMIS (Feustel and Rayner-Hooson 1990). These tools are tailored to handle air flows only, leaving it to the user to choose thermal boundary conditions.

The interaction between mass transports (air or water) and heat transfer in buildings is frequently significant. Thus, simulations of coupled systems are of interest, but, so far, fairly little work has been done in this field. For instance, (Hensen 1991) reports on such work using the ESP environment, but overall there has been a lack of adequate software for this type of studies.

Since the mid eighties, general, object oriented environments for building simulation are under development and in various stages of completion, e.g. CLIM 2000, EKS, SPARK, ZOOM. One important aim of these environments is to facilitate studies of coupled buildings and systems. The new tools will offer a unified handling of all component models, and thus allow simultaneous simulation of subsystems that hitherto have been handled by different programs. The general tools will normally be computationally less efficient than the specialised tools replaced; almost universally this drawback will be outweighed by the gain in man time efficiency.

Using the modern simulation environment IDA, a model family has been developed for studies of coupled air flows and heat transfers in multi-zone buildings. In addition to the coupling issues, this development has a special interest in that it offers an opportunity to weigh the pros and cons of general simulation tools versus application oriented ones, particularly with respect to: efficiency, robustness and development man-time. In this case, the application area is extremely well suited for tailored methods, so the odds against the general tools are high.

Dept. of Building Services Engineering,
Royal Institute of Technology, 100 44 STOCKHOLM
phone: +46-8-11 32 38, fax: +46-8-11 84 32,
e-mail: abring or plurre@engserv.kth.se

IDA has been developed at the Swedish Institute of Applied Mathematics in cooperation with the department of Building Services Engineering at KTH. At the heart of the system is a general solver for differential-algebraic (DAE) systems of equations, IDA Solver, which among other things features input-output free, precompiled component models. A discussion of solver design issues can be found in (Sahlin and Bring 1991) and (Eriksson, Söderlind, and Bring 1992) treats the numerical methods of IDA in some more detail. IDA Modeller is an interactive front end to the solver. IDA component models are primarily described with the Neutral Model Format, NMF (Sahlin, Bring, and Sowell 1992).

The present project has involved use of the entire IDA environment. The programming tool kit of IDA Modeller has been used to complement the general model-lab facilities of IDA. The result is a tailored user interface, focusing on the editing of the air flow networks and on specialised result presentations. These features of the current application, and the Modeller in general, are treated in an accompanying paper (Sahlin 1993). In this paper we will concentrate on NMF modelling of the components and on solver techniques.

The current application has been developed with support from the Swedish ventilation manufacturer, ABB Indoor Climate. Their primary interest is to use it for clean room design. ABB had evaluated the Movecomp program (developed by one of the present authors in cooperation with M. Herrlin), and found the functionality adequate but the user interface slightly dated. The idea to modernise Movecomp was discussed, but, since the resulting product would still be a monolithic program with uncertain development potential, an alternative was sought. The alternative, to implement the Movecomp functionality in IDA and develop a tailored tool, was accepted.

Section 2 covers the basic mathematical modelling, which is concretised further in Section 3 with sample NMF models. Application performance and numerical features are discussed in Sections 4 and 5. Due to space constraints, the account is focused on the multizone air exchange (MAE) models alone. These models have been successfully tested in conjunction with building thermal models, but the main difficulties are all MAE model related. Hence, our focus here.

2. FEATURES MODELLED

The implemented air-exchange models follow closely the approximations made in the Movecomp program (Herrlin 1992). Macroscopic models are used, assuming complete mixing within each zone (node). Obviously, for clean room design, it would be advantageous to study flow patterns within zones, especially in cases when doors are temporarily opened. However, this would require coupling of CFD models to lumped parameter models for ventilation and infiltration components. Such a task, although technically feasible, is outside the scope of our current design tool effort.

Besides the basic pressure - mass balance, we model transports of enthalpy plus one contaminant. Generally, the temperature distribution has a weak influence on the mass transport, whereas the contaminant is considered to lack such coupling. The current set of models treats only simple molecular transport of the contaminant. Sorption phenomena and molecular reactions are not included. It would be straightforward to extend the models to transports of more than one contaminant.

Excluding heat transfer models for the building envelope, we deal with two groups of air flow components; we will call them *nodes* and *connecting elements*.

The node components are characterised by their potentials: pressure (P), temperature (T), contaminant concentration (X). Nodes can be zones, or junctions in the ventilation system. Their model equations represent conservation of mass, energy, and contaminant:

$$\begin{cases} 0 = \sum \dot{m}_i \\ 0 = \sum q_i + q_{zone} \\ 0 = \sum xf_i + xf_{source} \end{cases} \qquad (1)$$

where q_{zone} is heat from the envelope and xf_{source} is a contaminant source term.

The connecting elements can be leaks, or any double-ended pieces of the ductwork (ducts, grilles, fans, etc.). Among their model equations we always find the mass flow modelled as a function of the pressure difference across the element,

$$\dot{m}_{1-2} = f(P_1 - P_2). \qquad (2)$$

We have chosen to use power law equations for the pressure - flow relation, rather than quadratic relations. Both choices have advantages; for us the main factor has been compatibility with Movecomp to simplify comparisons. For a leak, with linearization around zero, we then have,

$$f(\Delta p) = \begin{cases} c_0 \Delta p, & Abs(\Delta p) < \Delta p_0 \\ c(\Delta p)^n, & \Delta p > 0 \\ -c(-\Delta p)^n, & \Delta p < 0 \end{cases} \qquad (3)$$

where n is a coefficient between 0.5 and 1.0.

Heat and contaminant transport through a connecting element are typically just convected by the mass flow,

$$q_{1-2} = \begin{cases} c_p T_1 \dot{m}_{1-2}, & \dot{m}_{1-2} > 0 \\ c_p T_2 \dot{m}_{1-2}, & \dot{m}_{1-2} < 0 \end{cases} \text{ and} \qquad (4)$$

$$xf_{1-2} = \begin{cases} X_1 \dot{m}_{1-2}, & \dot{m}_{1-2} > 0 \\ X_2 \dot{m}_{1-2}, & \dot{m}_{1-2} < 0 \end{cases}, \qquad (5)$$

where T and X are the temperatures and contamination concentrations of the connected nodes on each side of the leak.

Heating or cooling coils will also model a local heat balance; a filter will model a reduction in contaminant concentration.

One of the major advantages with the modular approach is that replacement of individual component models is easy as long as the interface variables between models remain the same. This makes it possible to replace these simple models by more detailed ones. Thus, we could for instance introduce heat transports between ducts and surroundings, or detailed models of sorption phenomena. So far no such models have been implemented.

3. NMF AND EXAMPLES

In this section we will explain some basic features of the Neutral Model Format (NMF), and present NMF code for two representative models.

3.1. The Neutral Model Format (NMF)

NMF is a suggested standard for model expression. It has two main objectives: (1) models can be *automatically translated* into several simulation environments, i.e. the format is program *neutral* and machine readable; and (2) models should be easy to understand and express for non-experts. The first objective enables development of common model libraries, which can be accessed from a number of simulation environments.

Internal component model behaviour is described by a combination of algebraic and ordinary differential equations. (The given examples have only algebraic equations.) Equations may be written in any order and in the form

`<expression> = <expression>;`

NMF only *states* equation models, while *solution* of equations is, in some cases, left to the target environment (e.g. IDA, or SPARK), or the NMF translator in others (e.g. TRNSYS, or HVACSIM+).

NMF supports model encapsulation through a link concept, i.e. models may only interact via variables appearing in LINK statements. To enhance and encourage model plug compatibility, links and variables are globally typed.

To enable direct model translation to input-output oriented environments (e.g. TRNSYS, or HVACSIM+), variable declarations have a role attribute indicating IN for given variables and OUT for calculated ones. In addition, pure help variables, with role LOC, may be *assigned* to.

A complete account of NMF is given in (Sahlin, Bring, and Sowell 1992).

3.2. Zone Model Example

```
CONTINUOUS_MODEL BdZone
ABSTRACT
 "A static zone model for air-exchange modelling. Bidirectional
transports of energy plus a mass fraction are modelled."
EQUATIONS
/* mass conservation (eqn. 1a)*/
  0 = M_0 + SUM i=1, n M[i] END_SUM;
/* energy conservation (eqn 1b) */
  0 = Q_zone + Q_0 + SUM i2=1, n Q[i2] END_SUM;
/* fraction conservation (eqn. 1c)*/
  0 = xf_source + Xf_0 + SUM i3=1, n Xf[i3] END_SUM;
LINKS
/* type    name       variables... */
BidirX terminal_0 P, POS_IN M_0, T, POS_IN Q_0, X, POS_IN Xf_0;
FOR i = 1, n
 BidirX terminal[i]   P, POS_IN M[i], T, POS_IN Q[i], X, POS_IN Xf[i];
Tq      air_temp      T, POS_IN Q_zone;
VARIABLES
 /* type         name    role [def   min  max]  description */
 MassFlow_u  M_0       OUT             "terminal 0 mass flow"
 MassFlow_u  M[n]      IN              "terminal i mass flow"
 Pressure    P         IN              "zone floor level pressure"
 HeatFlux     Q_0      OUT             "terminal 0 heatflux"
 HeatFlux     Q[n]     IN              "terminal i heat flux"
 Temp         T        IN              "zone temperature"
 FractFlow_u  Xf_0     OUT             "terminal 0 transport"
 FractFlow_u  Xf[n]    IN              "terminal i transport"
 Fraction_y   X        IN              "zone fraction"
 HeatFlux     Q_zone IN                "heat gain/loss in zone"
MODEL_PARAMETERS
/* type   name    min  max       description */
 INT        n       1  BIGINT    "Number of links minus one"
PARAMETERS
/*type       name  [def  min  max]  description */
Length        za                   "zone floor height relative to ground"
Length        h   2.4  SMALL  BIG       "zone height"
Area          a    10  SMALL  BIG       "zone floor area"
FractFlow_u xf_source              "Mass fraction source (or sink)"
END_MODEL
```

NMF code for the air-exchange zone model

The link type for bidirectional flow between models BidirX has six variables: pressure and mass flow, temperature and heat flow, contaminant concentration and contaminant flow. It might seem that either temperature or heat flow could be eliminated, and corresponding for the contaminant pair of variables. It turns out, however, that modelling of bidirectional flow in a modular framework precludes that simplification. A discussion of this topic is outside the scope of this presentation, but it centers around the need to have a fixed number of equations for any component, independent of flow directions (contemplate the change when a T-piece swaps from converging to diverging state).

Besides the BidirX links, which handle air-exchange interaction with neighbouring models, the zone model has a Tq (temperature, heat flux) interface. This interface is the bridge between the air-exchange family of models and the thermal models for the building envelope. Due to space constraints we will refrain from a presentation of these models here.

A special case of scaling is used for time related variables, i.e. flows and time derivatives. These occur related to different time units, chosen to fit the current simulation focus. E.g. thermal building models are typically used with hour as time unit but may also be combined with flow models and control models in order to study faster phenomena at seconds scale. To avoid trivial model duplications, the scaling differences can be expressed in the models by using a time unit specified by a parameter t_scale. This scale gives the length in seconds of the time unit, t_scale = 3600 for hours, etc.

The zone and leak models have the BidirX interface which allows bi-directional flow. However, in the interest of calculation time we have chosen to model only uni-directional flow in ventilation components. This means that ventilation models need only four link variables, rather than six for the BidirX links and, generally, only one equation (eqn. 2), instead of three. The four-variable link type is called VentX.

3.3. Zone Supply Terminal Model

The supply terminal model, VxSupT, acts as interface between the VentX and BidirX models. Consequently, it has one link of each kind. This model still needs three equations - although only uni-directional flow is allowed - since there is a BidirX interface present.

```
CONTINUOUS_MODEL VxSupT
ABSTRACT "Supply Terminal. Linear flow below limit 'dp0', and if LIN."
EQUATIONS
/*two help assignments, for local density and pressure drop */
  Rho := rho_20 * (20. - ABS_ZERO) / (T2 - ABS_ZERO) ;
  Dp := P1 - P2 + zr2 * G * Rho ;
/* power law mass flow equation,  eqn. (2) in paper */
/* LINEARIZE function explained in Section 5.1 of paper */
  0 = -M / t_scale +
          IF LINEARIZE (1) THEN c_turb * Dp
          ELSE_IF Dp < dp0 THEN c_lin * Dp
          ELSE c_turb * sqrt (Dp)
          END_IF ;
/* convected heat through terminal, eqn. (4) in paper */
 Q = IF LINEARIZE (1) THEN T1
      ELSE cp * T1 * M / t_scale
      END_IF ;
/* fraction transported through terminal, eqn. (5) in paper */
 Xf = IF LINEARIZE (1) THEN X1
     ELSE X1 * M
     END_IF ;
LINKS
 /* type    name      variables... */
 VentX  inlet   P1, POS_IN M, T1, X1 ;
 BidirX  zone   P2, POS_OUT M, T2, POS_OUT Q, X2, POS_OUT Xf;
VARIABLES
/* type        name  role [def  min  max]      description */
 MassFlow_u M    OUT  0.  -BIG  BIG     "mass flow "
 Pressure    P1   IN    2.  SMALL  BIG  "pressure in"
 Pressure    P2   IN    1.  SMALL  BIG  "pressure out"
 temp        T1   IN    15. ABS_ZERO BIG "temperature in"
 temp        T2   IN    15. ABS_ZERO BIG "temperature zone"
 HeatFlux    Q    OUT   0.  -BIG  BIG "heat convected by massflow"
 fraction_y  X1   IN   .1   0.     BIG  "pollutant fraction in"
 fraction_y  X2   IN   .1   0.     BIG  "pollutant fraction zone"
 FractFlow_u Xf    OUT  0.  -BIG  BIG  "pollution transport"
 Pressure    Dp   LOC               "eff pressure diff"
 density     Rho  LOC               "air density"
PARAMETERS
 /* type      name [def   min   max] description */
/* easy access parameters */
/* priority order: c_t, xi */
 generic     c_t     0.  0       BIG   "power law coefficient"
 length      d      .25  SMALL  BIG   "inner diameter"
 generic     xi     10.  0       BIG   "loss coefficient"
 length      zr2     0.  0       BIG   "leak height from floor"
/* globally given */
 HeatCapM  cp   1006  500  3000     "air cp"
 Pressure   dp0  .1    SMALL  BIG   "limit for linear flow"
 Density    rho_20 1.2 SMALL  BIG    "reference density"
 Factor     t_scale 1.  SMALL  BIG    "size of time unit [s]"
/* derived parameters */
 area        a                         "cross section area"
 generic     c_lin                      "laminar coefficient"
 generic     c_turb                      "flow characteristic"
PARAMETER_PROCESSING
/*par processing is executed once, prior to simulation */
 a := PI * d * d / 4. ;
/* Check alternative definitions of C_turb */
 c_turb := IF c_t != 0. THEN
             c_t
           ELSE_IF xi == 0. THEN
             0.
           ELSE
             a * (2. * rho_20 / xi)**0.5
           END_IF ;
 IF c_turb == 0 THEN
 CALL nmf_error ("wrong parameters (c_t or xi) for supply terminal");
 END_IF ;
 c_lin := c_turb / sqrt (dp0) ;
END_MODEL
```

The use of the LINEARIZE function enhances model robustness. This issue is treated at some depth in Section 5.1.

4. PROBLEM SIZE

A crucial issue in our case of general versus specialised programs is obviously the question of overall efficiency.

In tailored programs for multi-zone air exchange, it is natural to reduce the central pressure - mass flow equation system (eqns 1 and 2) to contain just the flow balances in the nodes. The equations for the connecting elements relating pressure and flow (eqn. 2) are used to calculate the Jacobian matrix of the system with regard to the node pressures. The resulting linear system is positive definite and can therefore be solved without pivoting. Newton iteration, often damped, is used to solve the nonlinear system. The weak coupling from temperature to pressure, due to stack effect, can be handled in the same iteration loop without untoward effect on the convergence properties. The contaminant distribution, which normally has no feedback on pressure, can be calculated separately once the mass flows are known.

In a general simulation environment, such as IDA, the above type of simplification is not readily available. One of the aims of the current study has also been to demonstrate the applicability of a general tool on this type of problem. Thus, no effort has been made to utilise the special structure of the problem. The general sparsity techniques implemented in IDA are of course used. These are, on the other hand, most effective for problems with large components (many equations) but relatively few connections between components, and are thus not particularly well suited for the current model family.

In IDA, the equation system that is simultaneously solved will contain, both the conservation equations from the nodes, the pressure - flow relations from the connecting elements, and all transport equations. The system matrix will thus often be more than an order of magnitude greater than in the tailored program, especially when a ventilation system with many components is included.

At first sight this growth in system size seems, of course, fatal. However, for an overall appraisal of the approach, one should weigh the time spent on modelling, i.e. connecting models and giving parameters, against the raw simulation (solution) time. When IDA Modeller is used to set up a simulation problem, a reasonably sized problem may take about an hour to assemble, provided all parameters are known in advance. Let's say a problem with a three story building, five zones on each floor, and a balanced mechanical ventilation system. The time spent on actual number crunching on such a problem, with the general approach, is about half a minute on a 486 system. Obviously, the calculation time of a tailored program is only a few seconds, but this is, in our view, of little practical consequence, since the total turnaround time will be virtually the same. If a large number of simulations have to be done with the same model - for, say, automatic optimisation - the performance difference naturally becomes more important.

5. NUMERICAL APPROACH

In the previous section we have discussed the efficiency of the general tools in comparison with specialised. Naturally, the other crucial performance factor is robustness. If a general approach is significantly less reliable, it will clearly not be useful as a base for end user tools. Again, the multizone air-exchange set of models is extremely demanding.

Looking again at the tailored programs, we find that the restricted problem type can be exploited to enhance efficiency. Movecomp, e.g., uses a two step strategy: First, the flow equations are linearised. The exponents in the power law equations are set to one, and this linear system is solved. Secondly, starting from the linear solution, a modified newton method is used; in each step a Jacobian is calculated and a line search is made in the direction defined by a newton step. A formal proof has been found for the universal convergence of this method (Lindberg 1985) for the case of power law leak models. This is, indeed, quite remarkable since few such proofs exist for severely nonlinear systems.

For general DAE simulation, a separate initial value calculation is required to find start values satisfying the algebraic equations. For nonlinear systems, this task is a crucial problem, and will remain to be so, since no general solution is possible. Independent of problem type, the task can be solved if the user supplies good enough first guesses. Obviously, this can be very difficult for complicated models. Several general methods are available in IDA to deal with this problem:

- explicit linearization of NMF models to get a user independent starting point for nonlinear equation solving;
- two Newton homotopy techniques plus a line search in the Newton direction;
- a gradient solution method has been implemented and is under testing for cases when user guesses result in a singular Jacobian matrix.

The present version of the air-exchange library of models is completely static (algebraic), i.e. the initial value problem is the *whole* problem. However, for coupled models, e.g. when thermal models are included, we have a true DAE problem, but the initial value difficulties of this problem come from the air-exchange models alone. Hence, it is sufficient to look at them alone for experimentation with initial value techniques.

Two of the initial value tools of IDA have been developed in conjunction with the present project: explicit linearization and line search techniques.

5.1. Linearised NMF Models

Linearization of nonlinear models can be a useful means to support initial value calculation, quite independent of application field. We have thus chosen to make the linearization an explicit feature of the NMF models. This has been a planned extension of the NMF syntax for some time, but the present models are the first, where such a feature is indispensable. NMF-translators for various simulation environments may implement different interpretations, the simplest being to ignore the construction by implementing a dummy LINEARIZE function. This type of extension is however of general interest and we report the experience of our experiments so far.

The IDA solver is prepared to pass through a sequence of one or more preparatory stages at the beginning of each initial value calculation. Typically, there is only one extra stage, during which some components may choose to linearise their models. The stage information is requested by the component via a call of a Boolean system function

```
LINEARIZE (n),
```

where n is an integer constant. The function definition is:

LINEARIZE is true if the stage number of the solver is less than or equal to n.

The solver will let the stage number vary 1, 2, ... until no call of LINEARIZE gets an answer true. At that stage, all linearizations have been removed, and, if the solving has converged so far, one set of initial values has been found.

In the current application, all power law equations governing mass flow have been equipped with a linearised alternative, e.g. the leak mass balance (eqn. 2 and 3) is:

```
/*Leak with bidirectional flow */

/* power law mass flow equation */
 M / t_scale =
      IF LINEARIZE(1) THEN c * Dp
      ELSE_IF abs (Dp) < dp0 THEN c0 * Dp
      ELSE_IF Dp > 0 THEN c * Dp**n
      ELSE -c * (-Dp)**n
      END_IF ;
```

The equations governing heat flow and contaminant flow, being without major influence on the mass transport, can be manhandled even more. In a converging T-piece we might define the heat balance through:

```
/*Converging T-piece*/

/* energy balance equation */
0 = IF LINEARIZE (1) THEN
              - T3 + T1 + T2
          ELSE
              - M3 * T3 + M1 * T1 + M2 * T2
          END_IF ;
```

It is conceivable that a two stage relaxation of linearizations might be useful for this type of components. It could for instance help to keep temperature and contaminant equations linearised while the nonlinear mass flow equations are introduced. So far, this has not been called for in the current application.

For the air-exchange models the linearization technique relieves the user completely of guessing initial values. This is, in our view, an essential factor in the case of general vs. special.

5.2. Line Search Technique

Ordinary Newton-Raphson iteration is not a reliable tool for initial value calculation in nonlinear applications. Damped Newton-Raphson, is a better alternative, especially if the dampening factor is dynamically selected, e.g. by a line search. One such method has been implemented in conjunction with the current application.

A search is made in the direction defined by a newton step. The location of a minimum for a residual function is estimated, using quadratic interpolation between three suitable points. The minimised function is a weighted Euclidean norm of the residuals in the model equations:

$$\min_{\lambda} \sum_{i} (w_i * r(x_0 + \lambda * dx)_i)^2 \ , \qquad (6)$$

where **r** is the residual vector and **dx** defines the newton direction. **w** is a weight vector introduced to compensate for variations in scaling between the equations 0 = **F**(**x**):

$$w_i = 1 / \max_{k} \left(\left| \frac{\partial F_i}{\partial x_k} \right| \right) \qquad (7)$$

The mentioned explicit linearization and line search techniques have resulted in sufficient convergence properties for all practical purposes. Occasionally, for cases without any driving forces, the linearized solution leads to a singular Jacobian. However, these problems are likely to be solved with the recently implemented gradient methods.

5.3. Utilising Sparsity

A general problem formulation often leads to sparsely populated equation systems. The MAE models are, as we have seen, no exception in this respect. Consequently, a lot of the work on solvers for general simulation environments deals with various methods to utilise this sparsity with little or no loss of generality. IDA Solver provides several methods for various problem types, but many more could be developed. The numerical methods of IDA are presented at some depth in (Eriksson et al 1992). Here we will be content with a very superficial discussion.

The most sophisticated IDA methods, the *modular methods*, starts with a very large system of equations, which not only includes all component equations but also equations for every connected pair of variables, e.g.,

$$zone7.m_5 = -leak1.m_{1-2}. \qquad (6)$$

This class of methods is best suited for models with few component interface variables in comparison to internal variables. Another IDA method, the *compact method*, starts with eliminating the connection equations and then proceeds with solving the system without further utilisation of sparsity.

On the mentioned three-story test case, the best modular method outperforms the compact method slightly, in spite of the large number of interface variables.

The most obvious improvement would therefore be to implement some staple sparse techniques within the compact method, e.g. band and skyline sorting and solution. However, we believe a lot more can be done, and that, in fact, the MAE models represent a quite important problem category, i.e. a central mass flow - pressure system (eqns. 1 and 2) and several loosely coupled transport equations (eqns. 4 and 5). Topologically, all processes are spread across the network. Transports in the present models are only a single contaminant fraction and thermal energy, but they could be many more. Hitherto, we have not had the means to work on sparse methods for this class of problems but it seems evident that much could be done.

6. CONCLUSION

An application tool for inter-zone air exchange has been developed, using NMF and the general simulation tool IDA. NMF has proven a very effective means to describe component models, for human readers as well as for automatic translation into a simulation environment. IDA has been used to generate a tailored user interface for the application, making system building and manipulation simple tasks. The IDA Solver has proved to be an adequate tool for solving the nonlinear algebraic systems generated by the current application.

The application tool is presently under evaluation in an industrial design setting at ABB Indoor Climate.

A comparison with an existing simulation program, developed explicitly for inter-zonal air flow studies only (Movecomp), shows that the general approach is competitive:

- The tailored program is faster, but calculation times with the general program are quite acceptable also for fairly large systems, in spite of the facts that the problem type is extremely well suited for specialised solution methods and that the speed-up potential of the general tools is far from exhausted.

- The robustness of the general method is sufficient for practical purposes.

- Development time in the general environment is several times shorter than the specialised ditto.

- Model coupling with ,e.g., thermal models is straightforward in the general case while hardly practical in the specialised.

- The general models have significantly better maintenance and development potential. Adding a new model to the library only involves *formulating* the NMF model; the implementation time is negligible.

In summary, considering the continuous improvement of computer hardware, the weakness of the general system in calculation time is far outweighed by other factors. The same argumentation should be valid for most other building simulation applications, since the MAE models are rather demanding.

REFERENCES

Bring, A; Herrlin, M. 1991. Bris Data AB, Calscand International. *User's Manual, MOVECOMP-PC, An Air Infiltration and Ventilation System Program.*

Eriksson, L; Söderlind, G; Bring, A. 1992. "Numerical Methods for the Simulation of Modular Dynamical Systems." Bulletin 21. Dept. of Building Services Engineering, Royal Institute of Technology, Stockholm.

Feustel, H. E; Rayner-Hooson, A. 1990. *COMIS Fundamentals.* Lawrence Berkeley Laboratory, CA, USA.

Hensen, J.L.M; Clarke, J.A. 1991. "A Simulation Approach to the Evaluation of Coupled Heat and Mass Transfer in Buildings." In *Proceedings of the IBPSA Building Simulation '91* (Nice, Aug.).

Herrlin, M. 1992. "Air-Flow Studies in Multizone Buildings." Bulletin 23. Dept. of Building Services Engineering, Royal Institute of Technology, Stockholm.

Lindberg, B. 1985. "An Algorithm for Simulation of the Pressure Distribution in a Building." Research Report TRITA-NA-8503. Dept. of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm.

Sahlin, P; Bring, A. 1991. "IDA Solver - a Tool for Building and Energy Systems Simulation." In *Proceedings of the IBPSA Building Simulation '91* (Nice, Aug.).

Sahlin, P; Bring, A; Sowell, E. 1992. "The Neutral Model Format for Building Simulation." Bulletin 24. Dept. of Building Services Engineering, Royal Institute of Technology, Stockholm, Sweden.

Sahlin, P. 1993. "IDA Modeller - a Man-Model Interface for Building Simulation." To appear in *Proceedings of the IBPSA Building Simulation '93* (Adelaide, Aug.).

Walton, G. N. 1989. *AIRNET - a Computer Program for Building Airflow Network Modelling.* U.S. Department of Commerce, National Institute of Standard and Technology, Gaithersburg, MD, USA.