# AUTOMATIC BINDING OF SPICE MODELS TO THE DEVICE/CIRCUIT SIMULATOR MINIMOS-NT

Alexander Wolf, Tibor Grasser, and Siegfried Selberherr
Institute for Microelectronics
TU Wien
Gußhausstraße 27–29,
Vienna, Austria
E-mail: wolf@iue.tuwien.ac.at

## KEYWORDS

Computer Aided Engineering (CAE), Electronics, Combined simulation, Program generators, Simulation interfaces

## ABSTRACT

Numerical device simulation has proven to be an invaluable tool in the design and optimization of modern semiconductor devices. For the evaluation of device performance under realistic boundary conditions, mixed-mode device/circuit simulation programs like MINIMOS-NT have been developed. To make use of the vast number of compact models developed for SPICE, an automatic source-code based coupling procedure has been developed which generates code for MINIMOS-NT. A mixed mode device simulation is presented as an application of the new capabilities, which uses both the Berkeley BSIM3v3 model and distributed devices within one circuit.

## INTRODUCTION

Circuit simulation programs have in common that the electrical behavior of the devices is modeled by means of a compact model, that is, analytical expressions describing the device behavior. The most prominent program of this group is definitely SPICE. Once a suitable compact model is found, it can be evaluated in a very efficient way. However, this task is far from being trivial and many complicated models have been developed. To capture the features of modern ultra-small devices compact models have to be refined on a regular basis to meet the requirements of each new device generation. For the development and optimization of these devices numerical device simulation is frequently employed to circumvent the compact model development circle. As the behavior of a device under realistic boundary conditions can provide crucial information for the optimization task, solutions have been sought which combine circuit and device simulation capabilities.

One of these combined circuit/device simulators is MINIMOS-NT (Binder et al. 1998, Grasser 1999). Re-implementation of various compact models has proven to be a cumbersome task and therefore a different solution has been sought: as SPICE provides a fairly well defined interface which eases the implementation of new compact models many implementations of sophisticated models are available. These implementations should principally be reusable with a minimum of user interaction but the development of a suitable conversion tool proved to be a complex task.

## THE INTERFACE

As originally being a device simulator, MINIMOS-NT solves the constitutive relation formulation used for the semiconductor transport equations (Selberherr 1984). MINIMOS-NT solves $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ using a Newton algorithm

$$-\mathbb{J}^k \cdot \mathbf{u} = \mathbf{f}(\mathbf{x}^k) \quad (1)$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{u} \quad (2)$$

$$\mathbb{J}^k = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x} = \mathbf{x}^k} \quad (3)$$

where $\mathbb{J}$ is the Jacobian matrix, $\mathbf{u}$ is the solution vector of the linear equation system, also called update vector, $\mathbf{x}^k$ and $\mathbf{x}^{k+1}$ are the solutions of the previous and current iteration step, respectively. On the other hand SPICE uses the modified nodal approach (MNA) (Ho et al. 1975)

$$\mathbb{Y}^k \cdot \mathbf{x}^{k+1} = \mathbf{r}^k \quad (4)$$

with $\mathbb{Y}$ being the admittance matrix.

Substituting $\mathbf{u}$ from (2) into (1) yields a relation between the constitutive relation formulation and the MNA:

$$\underbrace{-\mathbb{J}^k}_{\mathbb{Y}^k} \cdot \mathbf{x}^{k+1} = \underbrace{\overbrace{\mathbf{f}(\mathbf{x}^k)}^{\mathbf{r}^k_{\text{MINIMOS-NT}}} - \mathbb{J}^k \cdot \mathbf{x}^k}_{\mathbf{r}^k_{\text{SPICE}}} \quad (5)$$

SPICE devices do not have access to the complete system matrix. They are in fact working on an empty subset of the system matrix which corresponds to the required number of nodes. The old right hand side is filled with the previous solution vector $\mathbf{x}^k$. With these values the models calculate and add their contributions to the admittance matrix $\mathbb{Y}^k$ which is

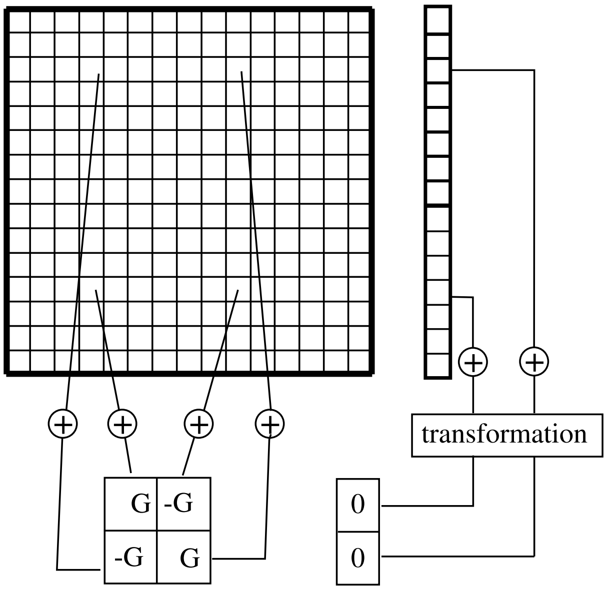
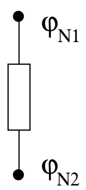



**Figure 1:** The resistor stamp super-positioned to the system matrix

super-positioned to the system matrix by the parent classes. Furthermore, the models add the new right hand side which has to be adapted via (5). As an example we consider the most simple device, a resistor.

The MNA-stamp of a resistor is given by:




| $y_{x,y}$ | $\varphi_{\mathrm{N1}}$ | $\varphi_{\mathrm{N2}}$ | J |
|---|---|---|---|
| $\mathrm{N}_1$ | G | $-$G | 0 |
| $\mathrm{N}_2$ | $-$G | G | 0 |

Fig. 1 shows how the admittances are super-positioned into the Jacobian according to the node numbers.
The right hand side has to be transformed and super-positioned to the system right hand side:

$$\mathbf{f}(\mathbf{x}^k) = \begin{pmatrix} \mathrm{f}_1(\mathbf{x}^k) \\ \vdots \\ \mathrm{f}_{N1}(\mathbf{x}^k) + \mathrm{G}\cdot(\mathrm{x}^k_{\mathrm{N1}} - \mathrm{x}^k_{\mathrm{N2}}) \\ \vdots \\ \mathrm{f}_{N2}(\mathbf{x}^k) + \mathrm{G}\cdot(\mathrm{x}^k_{\mathrm{N2}} - \mathrm{x}^k_{\mathrm{N1}}) \\ \vdots \\ \mathrm{f}_M(\mathbf{x}^k) \end{pmatrix} \tag{6}$$

with $\mathrm{f}_n(\mathbf{x}^k)$ being the sum of the right hand sides of all other devices but the resistor.

**Default Values of Input Parameters**

The default values for the SPICE models are hard-coded and thus not directly accessible to the user. This conflicts with the philosophy of MINIMOS-NT, which gives the user full access to the default values. Therefore a mechanism was implemented to choose if the devices should get default values from the MINIMOS-NT database, or if they should do the defaulting internally behaving exactly as SPICE.

**Inquiry of Output Parameters**

Access routines for all model and instance output parameters are provided even though only a small subset of them will be normally used.

**Providing Runtime Information**

To provide the runtime information for the SPICE models, several SPICE structures have to be properly allocated and updated, the most important ones represent the circuit structure and the admittance matrix respectively.

**Storing Time Dependent Data**

SPICE stores time dependent device data (e.g. junction capacitances) in a state vector for each device. This state vector is required for the trapezoidal integration and the higher order Gear integration algorithm. MINIMOS-NT, on the other hand, stores time dependent data on an equation system basis. Furthermore, only a simple backward Euler scheme is currently implemented for the semiconductor equations. Hence any additional information needs to be stored by the interface class.

**Implementation Issues**

One particular problem to solve was that MINIMOS-NT is written in C++ while SPICE is written in C. Hence a C++ wrapper-class had to be carefully designed to handle the communication between MINIMOS-NT and the SPICE devices in an abstracted way. This interface class is derived from the compact model base class offered by MINIMOS-NT. For each SPICE model an interface class is automatically generated which is itself derived from the SPICE base class. These interface classes manage device specific information like parameter names and types.

Although SPICE is written in C, an object oriented approach is used. Devices are splitted into models and instances of these models. The models contain parameters which a number of devices share like substrate doping or oxide thickness to mention just a few. On the other hand, device instances contain more specific parameters like channel length and other geometric specifications which are more likely to vary between individual devices. This separation is somehow arbitrary but reduces the size of the input files.

The approach taken in MINIMOS-NT is different. There,

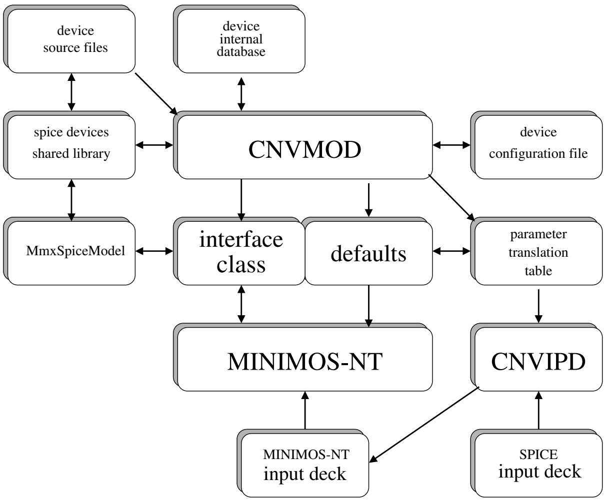



**Figure 2:** Conversion of a SPICE model

the input files provide the necessary object oriented features for a cleaner solution. For each device a base class is defined which contains all the available parameters together with appropriate default values. The instances of these devices occurring in the circuit are derived from this base class and modify the required values. An arbitrary number of intermediate classes may be created to capture the common features of the devices and to reflect the process under consideration. A MINIMOS-NT device does not distinguish between a model and instances of it, so a SPICE device instance is always merged with its model into a MINIMOS-NT device.

**CONVERSION OF MODELS**

Although it was intended to implement a fully automatic model converter, several design incompatibilities in the two simulators prevented this approach:

- The SPICE models do not always follow the naming conventions required by the spice implementation manual (Quarles 1989).
- MINIMOS-NT handles its parameters as consisting of values and units, a concept unknown to SPICE were default units are used throughout the input deck.
- While SPICE input-decks are parsed case insensitive, MINIMOS-NT's are case sensitive. Automatic creation of proper names might require user interaction.
- In MINIMOS-NT each parameter can be altered via a formula during simulation, taking runtime information like the current simulation time into account. Furthermore, each parameter may be stepped for the calculation of output curves. These features should also be available for SPICE model parameters which are normally initialized with constant values.
- The Input Deck database employed by MINIMOS-NT provides access to all parameters during the whole simulation. Output parameters like simulation results may be inquired via the `output` function. Unfortunately, some SPICE models modify their input parameters. Instead of modifying these values in the Input Deck database, the modified values may be obtained by a call to the `output` function.

Hence our approach was to implement a semi-automatic converter, which was called CNVMOD (convert models). Fig. 2 shows the main steps during model conversion:

**1.** In a first step all the binding of the SPICE source to CNVMOD takes place. The user can either provide the location of the main per-device functions[1] (CNVMOD then creates a VMAKE (Tuppa 1996) makefile entry) or it can be added by hand.

**2.** Afterwards a rebuild of CNVMOD is required, which also rebuilds a shared library containing the SPICE devices. This library will then be used by MINIMOS-NT and CNVMOD. Now CNVMOD has access to all parameters of the device via the standardized SPICE access routines.

**3.** In the next step a configuration file is automatically produced which can then be customized. A user defined name can be given to each terminal. In addition, each parameter

- can be assigned either an attribute type (e.g. resistance), if the attribute is known to MINIMOS-NT, or a unit string. The required information is normally shipped with the documentation of the device.
- can be made stepable and the update behavior can be specified. The update behavior is required when the parameter is calculated via a formula which depends on the internal state of the simulation. Possible update behaviors are: constant, update parameter value after each iteration or after each time step.
- can be altered in their input and output behavior as mentioned above.

Fig. 3 shows one parameter of the configuration file produced from the BSIM3v3 model. As can be seen from the automatically created comments, which are generated from the SPICE sources, the description is not always meaningful. The required information was found in the BSIM3v3 manual (Cheng et al. 1996) which stated that `ckappa` is the "coefficient for lightly doped region overlap fringing field capacitance", having the unit $\mathrm{F/m}$.

**4.** After customizing the configuration file, the interface class which consists of an include file and a C++ file is created by CNVMOD. A VMAKE makefile entry is produced in addition to a file containing the model default values ready to be added to the MINIMOS-NT default value database.

[1] As mentioned above naming conventions are sometimes not followed. Especially filenames are truncated to match the old MS-DOS 8.3 filename convention

```
// -------Description------
// New C-V model parameter
// Spice ID: 188
// -----------------------

ckappa
{
   MMNTname = "Ckappa";
   DataType = "Quantity";
   Unit     = "#F/m";
   Input    =  yes; Output = no;
   Stepping = "Constant";
}
```

**Figure 3:** Part of the configuration file obtained from the BSIM3v3 model, after user interaction.

**5.** Documentation in form of parameter tables showing the configurations made in a comprehensive way can be automatically generated, in either LaTeX, HTML, or plain text format.

**6.** At last a rebuild of MINIMOS-NT has to take place and afterwards the devices are ready to use in MINIMOS-NT.

**7.** After testing the device, an additional translation table file can be generated for the second tool CNVIPD (convert input decks) to get automatic conversions of SPICE input decks to MINIMOS-NT input decks.

## EXAMPLE: VOLTAGE-CONTROLLED OSCILLATOR

Fig. 6 shows a voltage-controlled oscillator proposed by (Väänänen et al. 2001). The circuit consists of a cross-coupled current-controlled oscillator and a controlling unit. The multivibrator is fed by a constant current source represented by $M_3$ and $M_4$. $Q_1$ and $Q_2$ together with the timing capacitor $C$ act as an emitter-coupled multivibrator, $Q_3$ and $Q_4$ form a differential pair directing the current through $C$. $Q_7$, $Q_8$, $M_1$ and $M_2$ function as active-pull-down output buffers. Transistors $Q_5$ and $Q_6$ direct the current which bypasses the timing capacitor to the controllable current sink formed by $Q_9$ and $Q_{10}$.

The rest of the circuit controls the current source output current and current sink input current, respectively. Controlled by the control voltage $V_{CONT}$ the two current sinks formed by $Q_{11}$, $Q_{12}$ and $Q_{13}$ drawn certain amounts of the reference currents. Hence the current through the multivibrator and the timing capacitor is controlled which defines the oscillating frequency.

The circuit was simulated using SPICE's modified Gummel-Poon model for the bipolar junction transistors and the BSIM3v3 (Cheng et al. 1997) for the MOS-transistors $M_3$ to $M_7$. To show MINIMOS-NT's capability of mixed-mode simulation and the mixing of SPICE devices with distributed devices, $M_1$ and $M_2$ were described using a distributed NMOS-model. The NMOS has a channel length of $L_G = 0.5\ \mu m$, the thickness is $W = 1\ \mu m$ and the oxide thickness is $d_{ox} = 3.5\ nm$. The substrate doping is $N_A = 10^{16}\ cm^{-3}$ and the source and drain maximum Gaussian contact doping level is $N_D = 10^{21}\ cm^{-3}$ The doping profile of the transistor is shown in Fig. 4.

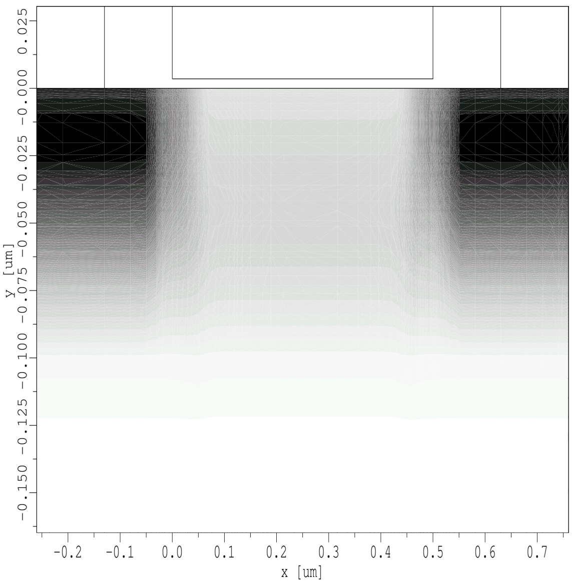



**Figure 4:** Doping profile of the distributed devices

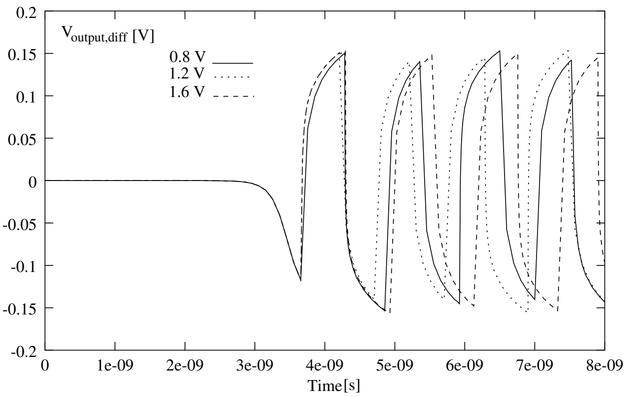



**Figure 5:** Voltage between the output nodes of the VCO at different control voltages

Fig. 5 shows the voltage between the output terminals for different voltages $V_{CONT}$. One can clearly see the voltage dependence of the frequency. It should be mentioned that the results differ from those found in (Väänänen et al. 2001). The VCO there has a center frequency at 4 GHz while the simulation shows a frequency at about 1 GHz. The differences can be explained by the lack of further knowledge of the devices being used in the Väänänen paper. Furthermore the goal was to show the possibility of simulating distributed and compact devices mixed in a circuit not the exact reproduction of the results.

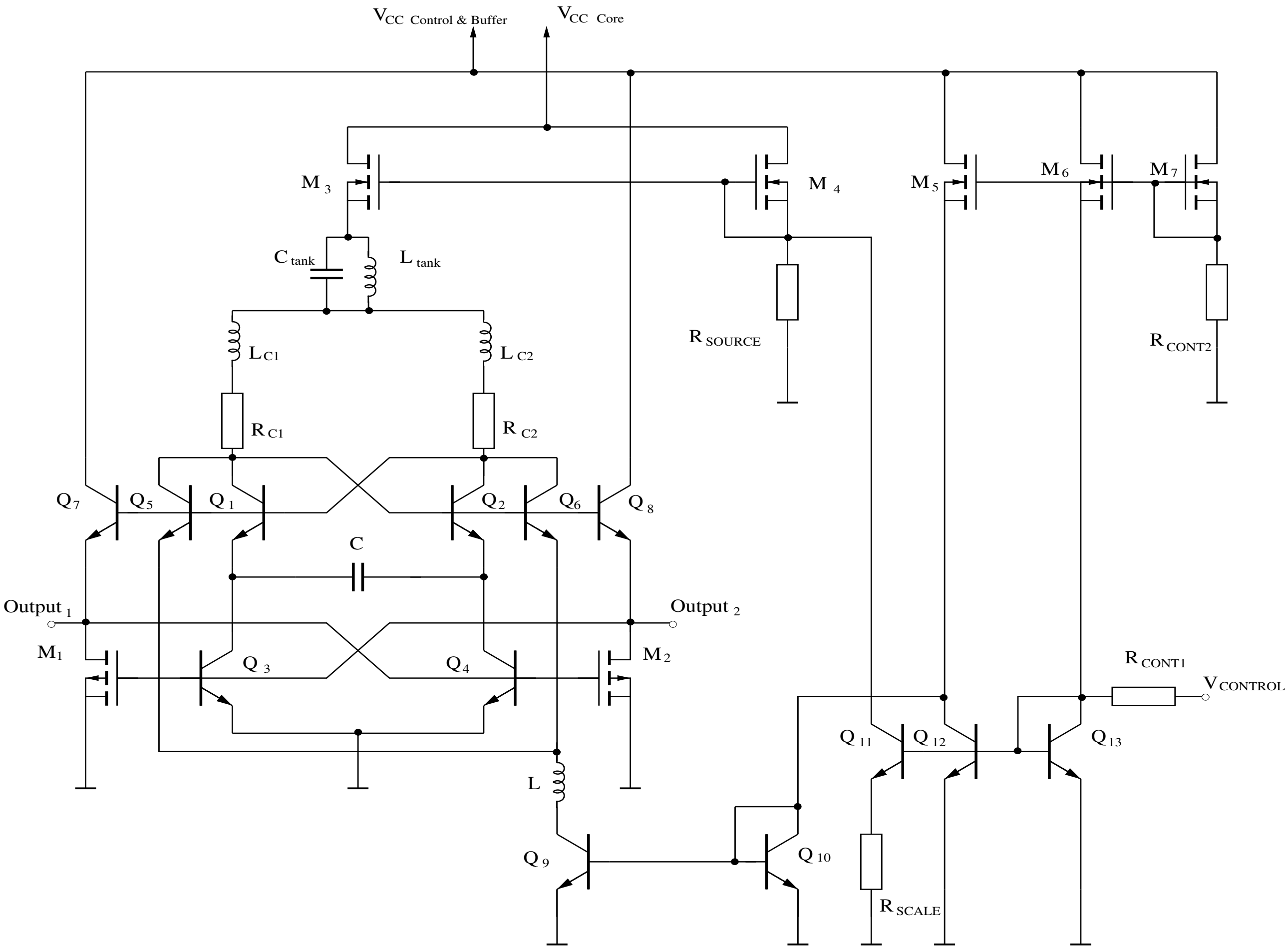



**Figure 6:** Voltage controlled oscillator

## CONCLUSION

We have implemented a tool for the semi-automatic conversion of source files written for SPICE compact models into their counterparts as required by the device/circuit simulator MINIMOS-NT. Although fully automatic model conversion has not been achieved, the advantages of this approach are obvious. The resulting overhead due to the wrapper functions and the right hand side recalculation is marginal when the models are used in context of a mixed mode device/circuit simulation as the distributed devices determine the complexity of the problem. In addition, a converter from SPICE to MINIMOS-NT input decks has been developed to ease testing and comparison between these two simulators.

## REFERENCES

Binder, T.; K. Dragosits; T. Grasser; R. Klima; M. Knaipp; H. Kosina; R. Mlekus; V. Palankovski; M. Rottinger; G. Schrom; S. Selberherr; and M. Stockinger, 1998, *MINIMOS-NT User's Guide*. Institut für Mikroelektronik.

Cheng, Y.; M.-C. Jeng; Z. Liu; J. Huang; M. Chan; K. Chen; P.K. Ko; and C. Hu, 1996, *BSIM3v3 Manual*. Department of Electrical Engineering and Computer Sciences, University of California, Berkeley.

Cheng, Y.; M.-C. Jeng; Z. Liu; J. Huang; M. Chan; K. Chen; P.K. Ko; and C. Hu, 1997, "A Physical and Scalable $I$-$V$ Model in BSIM3v3 for Analog/Digital Circuit Simulation". *IEEE Trans.Electron Devices*, 44(2):277–287.

Grasser, T., 1999, *Mixed-Mode Device Simulation*. Dissertation, Technische Universität Wien. http://www.iue.tuwien.ac.at.

Ho, C.W.; A.E. Ruehli; and P.A. Brennan, 1975, "The Modified Nodal Approach to Network Analysis". *IEEE Trans.Circuits and Systems*, CAS-22(6):504–509.

Klima, R.; T. Grasser; T. Binder; and S. Selberherr, November 1999, "Controlling TCAD Applications with a Dynamic Database". In *Software Engineering and Applications*, pages 103–112, Las Vegas, Nevada, USA.

Quarles, T.L., April 1989, The SPICE3 Implementation Guide. Technical Report UCB/ERL M89/44, Electronics Research Laboratory, University of California, Berkeley.

Selberherr, S., 1984, *Analysis and Simulation of Semiconductor Devices*. Springer, Wien, New York.

Tuppa, W., October 1996, *VMake – A CASE-Oriented Configuration Management Utility*. Dissertation, Technische Universität Wien. http://www.iue.tuwien.ac.at/diss/tuppa/diss-new/diss.html.

Väänänen, P.; M. Metsänvirta; and N. T. Tchamov, 2001, "A 4.3-GHz VCO with 2-GHz Tuning Range and Low Phase Noise". *IEEE J.Solid-State Circuits*, 36(1):142–146.