



## Original Research Article

### Development of a Computer-Based Cable Sizing Software for Domestic and Industrial Applications

**\*Dauda, A. and Omoze, L.E.**

Department of Electrical/Electronic Engineering, Faculty of Engineering, University of Benin, PMB 1154, Benin City, Nigeria.

\*abdul-rahman.dauda@uniben.edu; edirin.omoze@uniben.edu

<http://doi.org/10.5281/zenodo.21047921>

#### ARTICLE INFORMATION

##### Article history:

Received 18 May 2026

Revised 01 Jun. 2026

Accepted 06 Jun. 2026

Available online 30 Jun. 2026

##### Keywords:

Cable sizing

JAVA

Derating Factor

NetBeans IDE 8.2

Data Gathering

#### ABSTRACT

*Cable sizing is rigorous, as the different parameters needed are calculated manually through several standards and technical brochures. This work involves the development of a computer-based cable-sizing software for domestic and industrial applications. The software was developed using the JAVA programming language, and it consists of four major interfaces, namely circuit data, derating factor, cable characteristics, and results. The software works with IEEE Wiring Regulation, Nexans Kabelmetal Technical Datasheet, and Prysmian Technical Manual. These provide the data required for the calculation of the derating factors and information on the cable. These calculations were then compared with the acceptable standards already linked to the software to ensure proper cable sizing. The software was used to carry out calculations for domestic wiring, low voltage, and high voltage wiring. The results obtained were then compared with the manual's calculations to ascertain proper software operation. The software was found to be accurate, hence providing a fast and accurate means of carrying out the cable sizing, process which makes the design process easier.*

© 2026 RJEES. All rights reserved.

## 1. INTRODUCTION

Cables are assemblies of one or more insulated conductors used for the transmission of electrical power or data signals (Reddy, 2021). Correct cable selection is critical: an undersized cable risks damage and fire hazard, while an oversized cable is uneconomical (Olatomiwa and Akinyemi, 2012). Cable sizing calculations ensure safe operation under full load, adequate voltage supply, and correct operation of protective devices during faults (Nair and Nithiyanthan, 2016).

Internationally recognised standards govern this process, including the IEEE Wiring Regulations (BS 7671) and IEC 60364-5-52 (2009). Despite these standards, cable sizing remains a laborious task

involving multiple correction factors, catalogue lookups, and iterative calculations. This inefficiency motivates the development of dedicated software tools.

Existing cable sizing tools vary widely in cost, scope, and ease of use. Olatomiwa and Akinyemi (2012) developed a Java-based calculator for residential services but supported only one cable type. Nair and Nithiyananthan (2016) proposed an effective cable sizing model addressing current-carrying capacity, voltage drop, and short-circuit energy. Anders (1997) provided a comprehensive treatment of cable thermal ratings underpinning derating methodology, while Glover et al. (2012) and Tleis (2008) offer power system fundamentals relevant to impedance and voltage-drop analysis. IEC 60287 (2006) establishes the international standard for continuous current ratings of cables.

The present work describes a Java-based cable sizing application that integrates the relevant standards and manufacturer data into a single interactive tool for both domestic and industrial use. The software supports domestic, low-voltage (LV), and high-voltage (HV) applications and was validated against manual calculations.

## 2. MATERIALS AND METHODS

### 2.1 Cable Construction

Figure 1 shows the main components of a cable. A cable may have one or more cores depending on the service type. For example, a three-core cable is used for 3-phase service. Conductors are made of tinned copper or aluminium and are usually stranded for flexibility (Mehta and Mehta, 2008). Each core is provided with insulation of appropriate thickness for the voltage level. Common insulation materials include impregnated paper, varnished cambric, and rubber mineral compound (Mehta and Mehta, 2008). A layer of fibrous material (e.g. jute or hessian tape) called bedding is applied over the metallic sheath to protect it from corrosion and mechanical injury caused by armouring (Mehta and Mehta, 2008). The Armouring is made up of one or two layers of galvanised steel wire or tape called Steel Wire Armouring (SWA) or aluminium wire called Aluminium Wire Armouring (AWA) applied over the bedding (Shelton, 2005). SWA is used on multicore cables; AWA on single-core cables (aluminium being non-magnetic avoids induced currents).

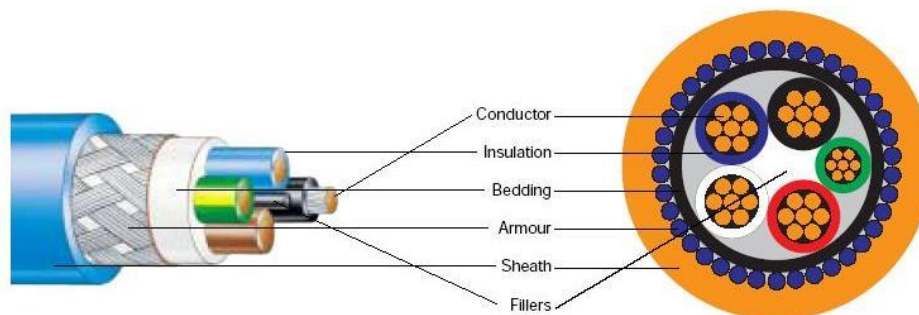


Figure 1: Parts of a Cable

Armouring protects the cable from mechanical damage during installation and handling, and may be omitted on some cable types. (Theraja and Theraja, 2017). A lead or aluminium sheath is applied over the insulation to protect the cable from moisture, gases, and corrosive liquids. Primarily used in outdoor and mains cables (Mehta and Mehta, 2008). A fibrous layer (e.g. jute) called serving is applied over the armouring to protect it from atmospheric conditions (Mehta and Mehta, 2008).

### 2.2. Design Method

The cable sizing application was developed in JAVA using NetBeans IDE 8.2. JAVA was selected for its platform independence across any Java Runtime Environment (JRE) (Oracle Corporation, 2020). Manufacturer data from the Nexans Kabelmetal Technical Datasheet (2005) and Prysmian Technical Manual (2010) were embedded in Microsoft Excel spreadsheets queried at runtime.

The software follows a strictly sequential design procedure consisting of five steps: (i) data gathering, (ii) full load current computation, (iii) derating factor calculation, (iv) cable selection from catalogue data, and (v) voltage drop and short-circuit verification (Open Electrical, 2018). These steps are implemented across four graphical user interfaces (GUI), as illustrated in Figure 2, and are described in Sections 2.2.1 to 2.2.4.

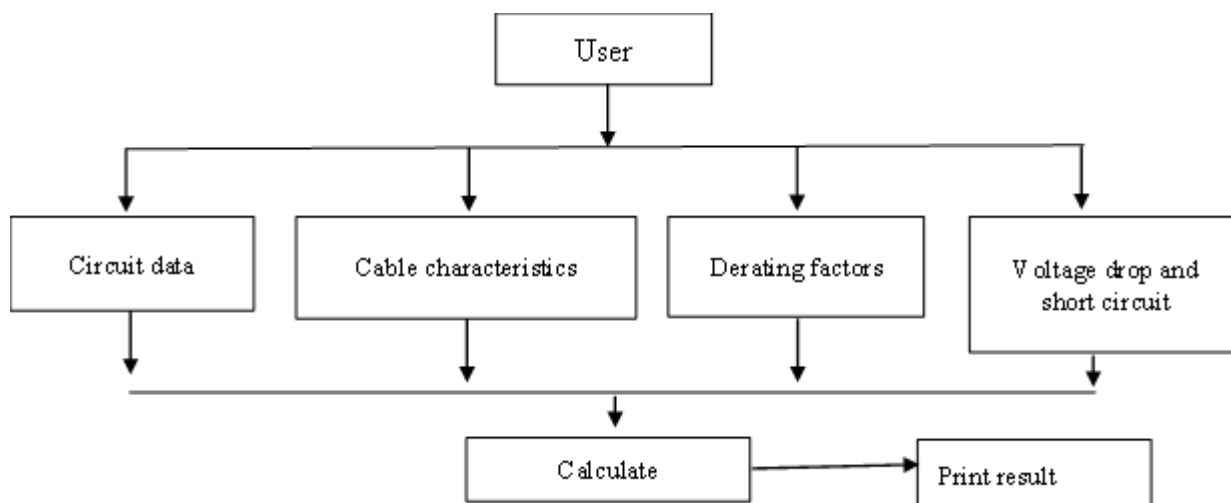


Figure 2: Sequence of operation for the cable sizing software

The user interacts with each interface by inputting the required data and can toggle freely between interfaces. Results can be printed as a technical document. Figure 3 shows a screenshot of the software.

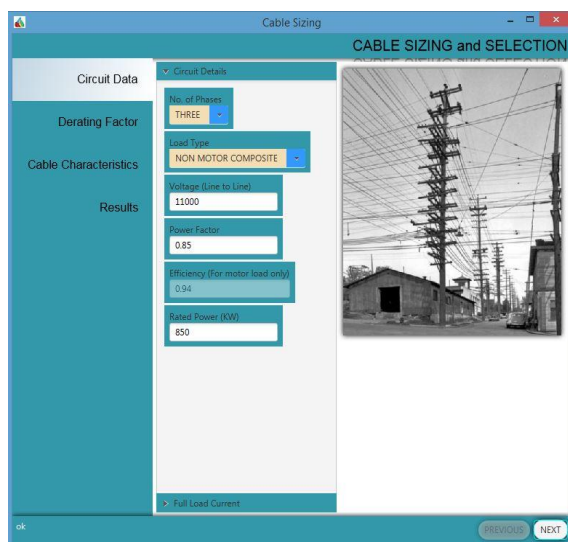


Figure 3: Circuit data interface

### 2.2.1. Circuit data interface

Step 1 of the design procedure is data entry and full load current calculation. The user provides the load type (motor or non-motor composite), number of phases, rated power (kW), supply voltage (V), power factor, and motor efficiency (for motor loads). If the full load current is already known, it may be entered directly, bypassing the computation. The CircuitData class implements this computation as follows. For a three-phase non-motor load, the full load current is given by Equation (1).

$$I_{flc} = \frac{P}{\sqrt{3} \times V \times \cos \phi} \quad (1)$$

$I_{flc}$  = full load current, V=Voltage, P= Real Power,  $\cos \phi$  =power factor.

The full load current for a single phase non motor load is given in Equation (2).

$$I_{flc} = \frac{P}{V \times \cos \phi} \quad (2)$$

$I_{flc}$  = full load current, V=Voltage, P= Real Power,  $\cos \phi$  =power factor.

The full load current for a three-phase motor load is given in Equation (3).

$$I_{flc} = \frac{P}{\sqrt{3} \times V \times \text{Eff} \times \cos \phi} \quad (3)$$

$I_{flc}$  = full load current, V=Voltage, P= Real Power,  $\cos \phi$  =power factor, Eff=Efficiency.

The full load current for a single-phase motor load is given in Equation (4).

$$I_{flc} = \frac{P}{V \times \text{Eff} \times \cos \phi} \quad (4)$$

$I_{flc}$  = full load current, V=Voltage, P= Real Power,  $\cos \phi$  =power factor, Eff=Efficiency.

These equations are implemented directly in the `getFullLoadCurrent()` method of the `CircuitData` class, as shown in the following code snippet:

---

```
// CircuitData.java — Full Load Current calculation
public double getFullLoadCurrent(boolean solve) {
    if (solve) {
        if (getNumberOfPhases() == Phases.THREE) {
            // Three-phase: Iflc = P / (√3 × V × PF × Eff)
            fullLoadCurrent = getRatedPowerKA()
                / (Math.sqrt(3) * getVoltageLtoL() / 1000);
        } else {
            // Single-phase: Iflc = P / (V/√3 × PF × Eff)
            fullLoadCurrent = getRatedPowerKA()
                / (getVoltageLtoN() / 1000);
        }
    }
    return fullLoadCurrent;
}

// getRatedPowerKA() accounts for motor efficiency and power factor:
public double getRatedPowerKA() {
    if (getLoadType() == LoadType.MOTOR_ONLY) {
        ratedPowerKA = getRatedPowerKW()
            / (getPowerFactor() * getMotorEfficiency());
    } else {
        ratedPowerKA = getRatedPowerKW() / getPowerFactor();
    }
    return ratedPowerKA;
}
```

---

The computed  $I_{flc}$  is stored in the `AppVariables.activeCircuitData` object and passed to the derating factor interface.

The calculated full load current is then divided by the derating factor, which is calculated depending on where the cable is to be laid (underground or air). The derated current is given in Equation (5).

### 2.2.2. Derating factor interface

Step 2 is the calculation of the composite derating factor  $K_D$ , which accounts for the reduction in current-carrying capacity caused by installation conditions. The interface is split into two sections depending on the cable laying method: above-ground (in-air) or underground. For cables installed in air, the derating factor is the product of the ambient temperature correction factor and the grouping factor per Equation (5). For cables laid in air the derating factor is given in Equation (5).

$$K_D = \text{Ambient temperature correction factor} \times \text{Grouping factor} \quad (5)$$

For underground cables, the derating factor incorporates additional soil and depth corrections per Equation (6).

$$K_D = \text{Thermal resistivity correction factor} \times \text{depth correction factor} \times \text{Ground temperature correction factor} \times \text{grouping factor} \quad (6)$$

The derated load current used for cable selection is then obtained from Equation (7).

$$I_D = \frac{I_{flc}}{K_D} \quad (7)$$

$I_D$  = Derated current,  $I_{flc}$  = full load current,  $K_D$  = Derating factor.

A cable size is then selected based on the derated current value from a cable catalogue or technical brochure.

The following snippet from the OverheadLineInfo class (in-air case) shows the implementation:

---

```
// OverheadLineInfo.java — In-air derating factor
public double getDeratingFactor(boolean solve) {
    if (solve) {
        // KD = ambient temperature factor × grouping factor
        deratingFactor = getAmbientTemperaturFactor()
            * getCableInGroupCorrectionFactor();
        return deratingFactor;
    } else {
        return deratingFactor; // return previously stored value
    }
}

// Correction factors are looked up from Excel datasheets
// via the DataManager class:
public double getCableInGroupCorrectionFactor() {
    if (cableCores == CableCores.SINGLE) {
        return cableInGroupCorrectionFactor =
            (double) cableInGroupModel.getValue(
                numberOfCablesInGroup, numberOfCablesTrays);
    } else {
        // multicore: apply additional correction for arrangement
        double corr = cableInGroupModel.getValue(
            numberOfCablesInGroup, numberOfCablesTrays);
        return cableInGroupCorrectionFactor =
            corr * multicoreCorrectionModel.getValue(
                row, numberOfCableCores - 1);
    }
}

// Derated current in CableParamsBase:
```

```

public double getDeratedLoadCurrent() {
    return fullLoadCurrent / deratingFactor; // Eq. (7)
}

```

The derating factor interface is shown in Figure 4.

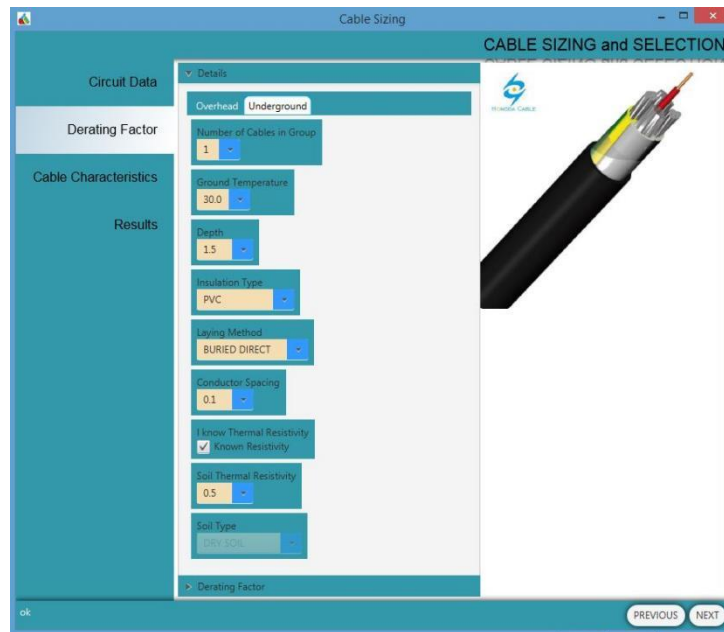


Figure 4: Derating factor interface

### 2.2.3. Cable characteristics interface

Step 3 is cable selection and verification. Using the derated current ID from Step 2, the software queries the pre-loaded Excel cable catalogue (via the DataManager class) to identify the smallest standard cable size whose current rating equals or exceeds ID. The CableParamsBase.getSuggestedCableSize() method performs this lookup using the getColumnIndexOrUpperOf() function, which traverses the rating column of the datasheet and returns the first entry at or above ID.

Step 4 is voltage drop verification. For low voltage cables, the voltage drop is given in Equation (8).

$$\Delta U = \frac{V_d \times L \times I}{1000} \quad (8)$$

where U is the voltage drop (V),  $V_d$  is the voltage drop per ampere per metre (mV/A/m) from the cable datasheet, L is the cable length (m),  $\Delta U$  is the voltage drop, and I is the full load current (A).

The percentage voltage drop is then calculated for three-phase in Equation (9).

$$\Delta U\% = \frac{\Delta U}{400} \times 100 \quad (9)$$

$\Delta U$  is the voltage drop and  $\Delta U\%$  is the percentage voltage drop.

And for single phase in Equation (10):

$$\Delta U\% = \frac{\Delta U}{230} \times 100 \quad (10)$$

$\Delta U$  is the voltage drop and  $\Delta U\%$  is the percentage voltage drop.

Calculating Percentage Voltage Drop for HV Cables

The voltage drop for high voltage cables is given in Equation (11).

$$\text{Voltage drop} = \frac{\sqrt{3} \times I_L \times L[Z]}{V \times 1000} \times 100 \quad (11)$$

$I_L$  is the Full Load Current (Ampere),  $L$  is the cable length (m),  $Z$  is the Cable Impedance (ohm) and  $V$  is Line Voltage (kV).

The percentage voltage drop for high voltage cables is given in Equation (12).

$$\% \Delta U = \frac{\text{Voltage drop}}{\text{Rated voltage}} \times 100 \quad (12)$$

If the computed  $U\%$  exceeds the permissible limit (2.5% for final circuits; 5% for mains cables per BS 7671), the software automatically increments the cable size by one step and repeats the check. This iterative selection is implemented in the `optimunResult()` method of `CableParamsBase`, shown below:

---

```
// CableParamsBase.java — iterative cable selection
public void optimunResult(boolean res) {
    if (!res) {
        offset = 0; // no optimisation; use minimum rated cable
    } else {
        if (loadType == CircuitData.LoadType.NON_MOTOR_COMPOSITE) {
            // Step up cable size until voltage drop is within limit
            while (getPercentageVoltageDrop() > getAllowedVoltageDrop()) {
                offset += 1;
            }
        } else {
            // Motor load: check both running and start-up voltage drop
            // Start-up current = 6 × Iflc (getPercentageVoltageDropForMotor
            while (getPercentageVoltageDrop() > getAllowedVoltageDrop()
                || getPercentageVoltageDropForMotor()
                > getAllowedVoltageDrop() * 4) {
                offset += 1;
            }
        }
    }
}

// LVCableParams.java — voltage drop calculation (Eq. 8)
@Override
public double getVoltageDrop() {
    return voltageDrop =
        (getVoltageDrop_per_A_per_m() * getCableLength()
            * getFullLoadCurrent()) / (1000 * getNumberOfCableRun());
}

@Override
public double getPercentageVoltageDrop() {
    return getVoltageDrop() / getVoltageLtoL() * 100; // Eq. (9)/(10)
}

@Override
public double getPercentageVoltageDropForMotor() {
    return getPercentageVoltageDrop() * 6; // 6× start-up current
}

```

---

Cable data is retrieved from the database to compute the percentage voltage drop. If the calculated drop exceeds the permissible limit, a larger cable size is automatically selected. The cable characteristics interface is shown in Figure 5.

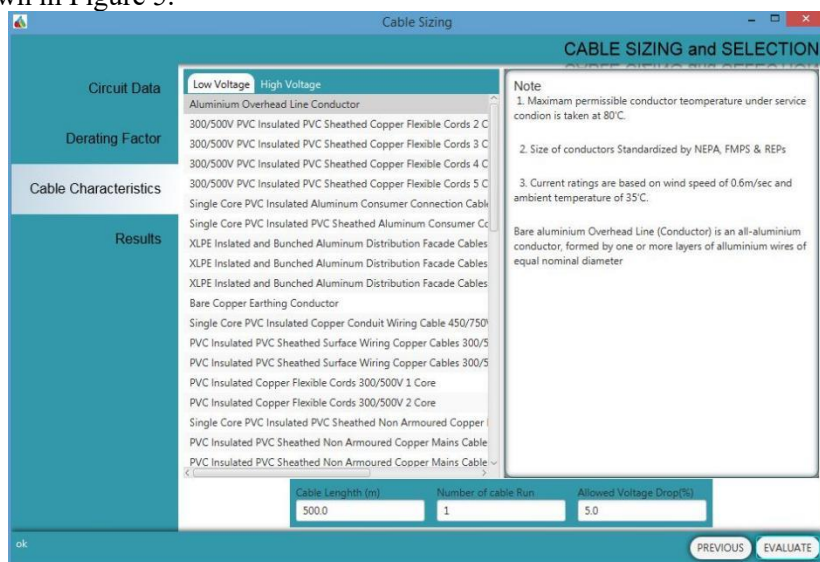


Figure 5: Cable Characteristics Interface

#### 2.2.4. Short-circuit verification

Step 5 is short-circuit verification. For new installations, the short-circuit current rating of the selected cable (kA for 1 s, from column 5 of the cable datasheet) must exceed the prospective fault current at the cable origin. For existing installations where the upstream protective device is known, the let-through energy  $I^2t$  of the device is compared with the cable's thermal withstand capacity  $k^2S^2$ , where  $k$  is the conductor material constant and  $S$  is the conductor cross-section ( $\text{mm}^2$ ). The short-circuit rating is retrieved from the datasheet via `getShortCircuitRatingInKA()` in `CableParamsBase`:

---

```
// CableParamsBase.java — short-circuit rating lookup
@Override
public double getShortCircuitRatingInKA() {
    setupDataModel();
    int row = getRow(offset); // row determined by derated current
    return suggestedCableSize = dataModel.getValue(row, 5);
    // Column 5 in the Excel datasheet holds the 1-second
    // short-circuit current rating (kA) for each cable size
}
```

---

#### 2.2.5. Results interface

This interface displays the selected cable size and all calculated parameters with explanatory text to assist users new to cable sizing. A print option allows results to be exported as a technical document. The results interface is shown in Figure 6.

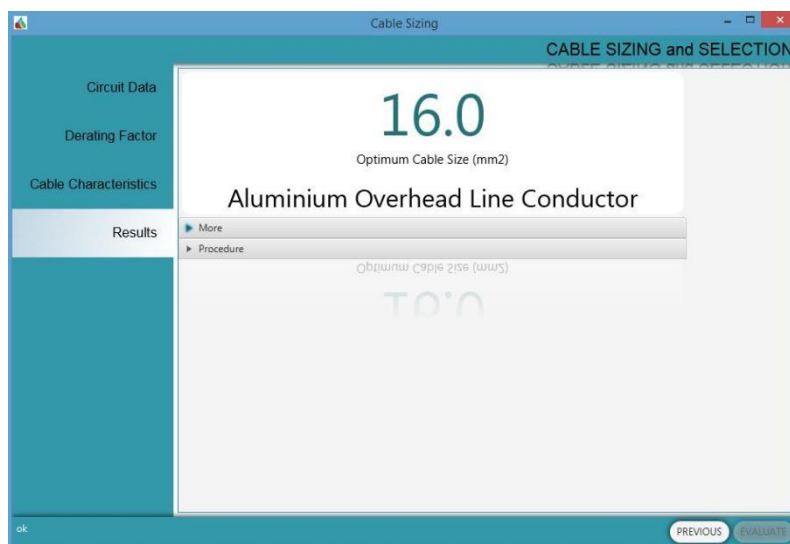


Figure 6: Results interface

### 2.3. Validation Approach

Three test cases spanning domestic, LV mains (motor), and HV mains applications were used to validate the software. For each case, the full load current, derating factor, derated current, cable size, percentage voltage drop, and short-circuit rating were calculated independently by hand using the same standards and manufacturer data embedded in the software. The software outputs were then compared with these manual results, and agreement was assessed on a parameter-by-parameter basis.

## 3. RESULTS AND DISCUSSION

### 3.1. Test 1: Indoor Wiring (Domestic)

Three test cases were used to validate the software against manual calculations. The results are summarised in Table 1, followed by detailed case analyses. A PVC insulated, twin 20 m cable supplied a 6000 W cooker with an integral socket outlet in domestic premises. Ambient temperature was 30°C; the cable was installed on a perforated cable tray.

Table 1: Summary of validation results

Parameter	Manual result	Software result
Test 1 – Cable Size	2.5 mm <sup>2</sup>	2.5 mm <sup>2</sup>
Test 1 – Voltage Drop (%)	1.63%	1.63%
Test 2 – Cable Size	3C × 50 mm <sup>2</sup>	3C × 50 mm <sup>2</sup>
Test 2 – Voltage Drop (%)	2.8%	2.8%
Test 2 – Start-up Voltage Drop (%)	17%	17%
Test 3 – Cable Size	3C × 25 mm <sup>2</sup>	3C × 25 mm <sup>2</sup>
Test 3 – Voltage Drop (%)	0.0146%	0.0146%

The full load current was calculated using Equation (2) as 25 A. Applying diversity for a cooker with socket outlet (Table 2), the design current was reduced to 19.5 A. A 2.5 mm<sup>2</sup> cable (rated 21 A) was proposed. The percentage voltage drop was computed using Equations (8) and (10): %U = 1.63%, which is within the 2.5% limit specified by BSI (2008). The short-circuit capacity of the 2.5 mm<sup>2</sup> cable was verified against the upstream protective device.

Table 2: Cooker diversity calculation

Item	Current (A)
First 10 A	10
30% of remainder (30/100 × 15)	4.5
Socket outlet allowance	5
<b>Total Design Current</b>	<b>19.5</b>

**Result:** 1 run of 2.5 mm<sup>2</sup> LV cable selected. The software output (Figures 7–10) matched the manual calculation exactly.

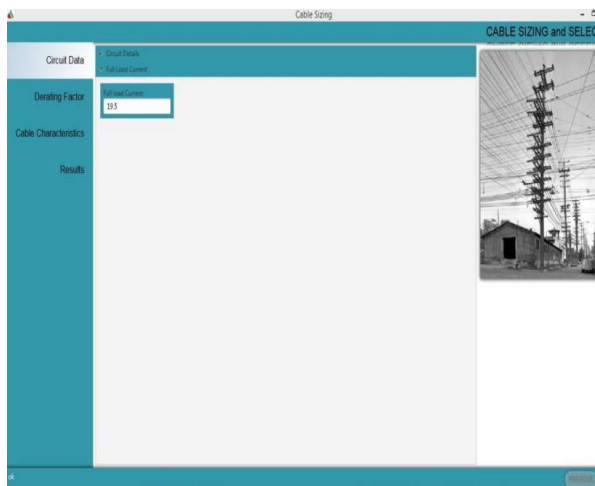


Figure 7: Circuit data interface for indoor wiring calculation



Figure 8: Derating factor interface for indoor wiring calculation

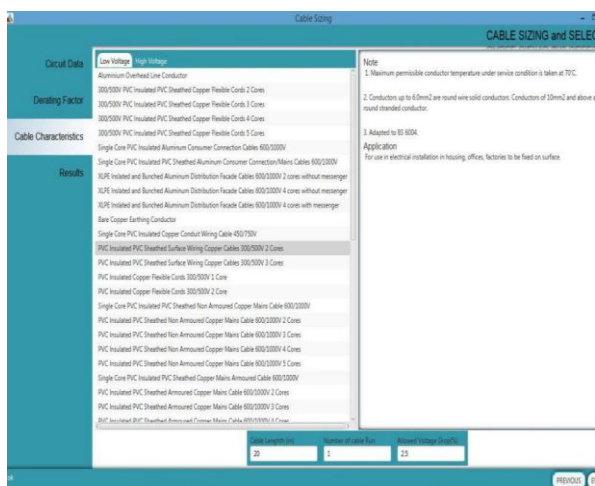


Figure 9: Cable characteristics interface for indoor wiring calculation

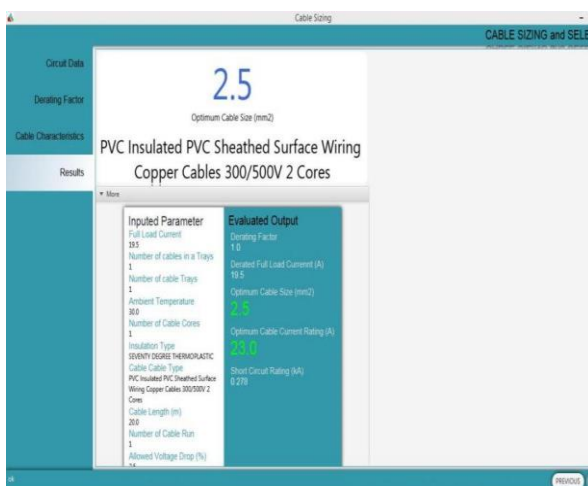


Figure 10: Result for indoor wiring calculation

**3.2. Test 2: LV Mains (Motor Load)**

A 400 V, 45 kW motor was supplied over a 160 m run, protected by a 200 A fuse. Motor efficiency was 94%, power factor 0.85. The cable was laid underground in a group of nine cables at a depth of 0.7 m in dry soil at 25°C. The full load current was computed using Equation (3) as 81 A. The derating factor was calculated from Equation (6) as 0.81. The derated current from Equation (7) was 100 A, requiring

a cable rated at  $\geq 100$  A. A  $3C \times 50$  mm<sup>2</sup> armoured LV cable was selected ( $V_d = 0.87$  mV/A/m). Applying Equation (8), the voltage drop is 11 V; by Equation (9), the percentage voltage drop is 2.8% which is  $< 5\%$  being the limit. For motor start-up the start-up current is 486 A, percentage voltage drop is 17% which is  $< 20\%$  being the permissible limit. Short-circuit capacity of the  $3C \times 50$  mm<sup>2</sup> cable (7.2 kA for 1 s) exceeded the let-through energy of the 200 A fuse ( $I^2t = 0.3 \times 10^6$  A<sup>2</sup>s;  $k = 143$ , per BS 7671 Table 54C).

**Result:** 1 run of  $3C \times 50$  mm<sup>2</sup> armoured LV cable selected. All software outputs (Figures 11–14) matched manual values.

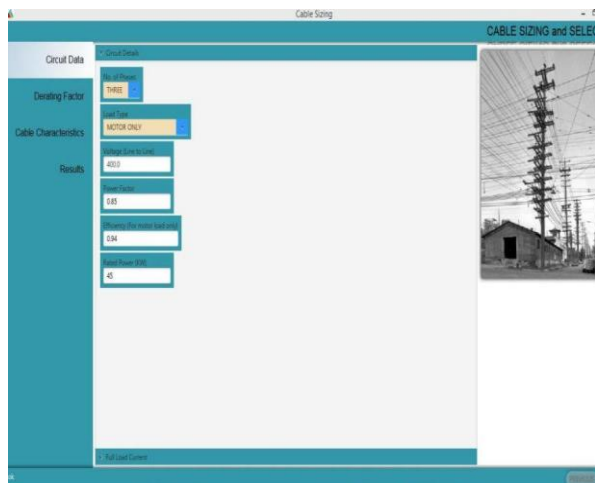


Figure 11: Circuit data interface for low voltage mains calculation

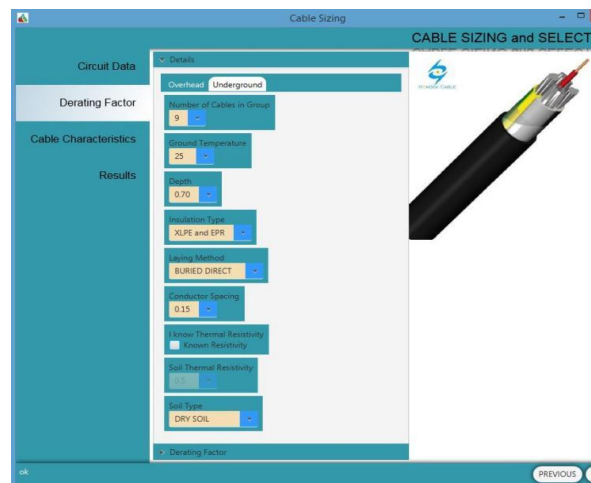


Figure 12: Derating factor interface for low voltage mains calculation

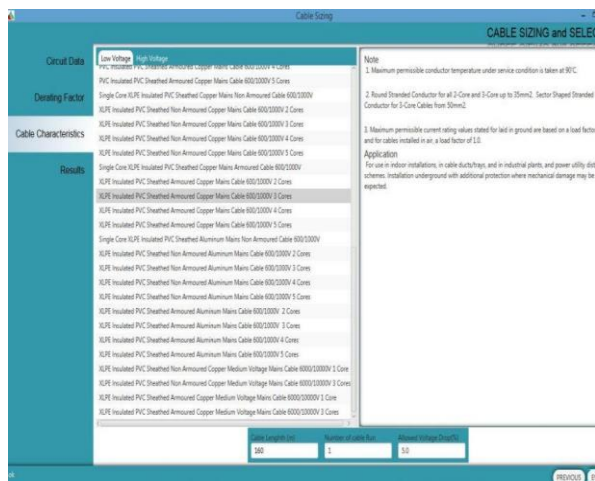


Figure 13: Cable characteristics interface for low voltage mains calculation

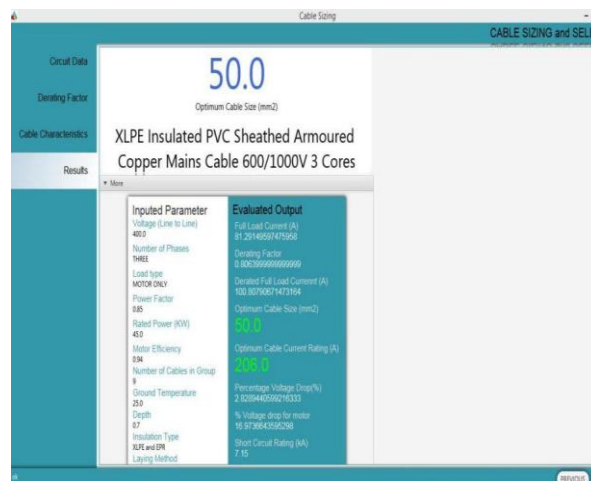


Figure 14: Result for low voltage mains calculation

### 3.3. Test 3: HV Mains

A three-phase 11 kV, 1000 kVA supply cable was sized for a 15 m run. The cable was laid underground in a group of nine cables at 1 m depth in dry soil at 35°C. From Equation (3),  $I_{flc} \approx 52.5$  A. Applying Equation (7):  $K_D \approx 0.73$ . From Equation (5):  $I_D \approx 72$  A. A  $3C \times 25$  mm<sup>2</sup> armoured HV cable (rated 135 A) was proposed. Using cable datasheet values and Equations (11)–(13) with  $L = 15$  m:  $\%U = 0.0146\% < 5\%$  limit.

**Result:** 1 run of  $3C \times 25 \text{ mm}^2$  armoured HV cable selected. The software output (Figures 15–18) was in complete agreement with manual calculations. The software results were consistent with all manual calculations. Completion time was significantly reduced compared to manual methods, owing to the speed advantage of automated computation.

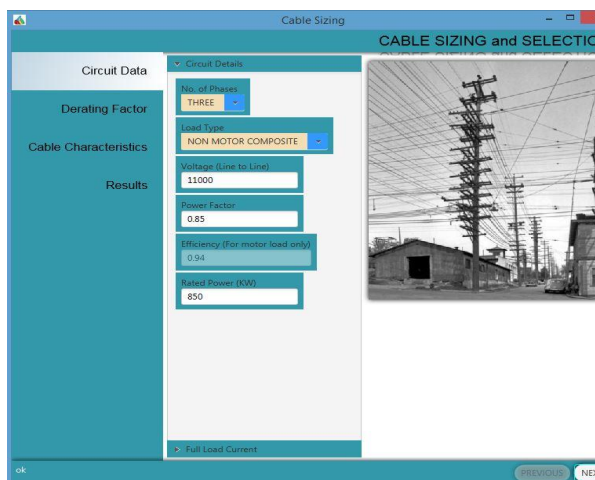


Figure 15: Circuit data interface for high voltage mains calculation

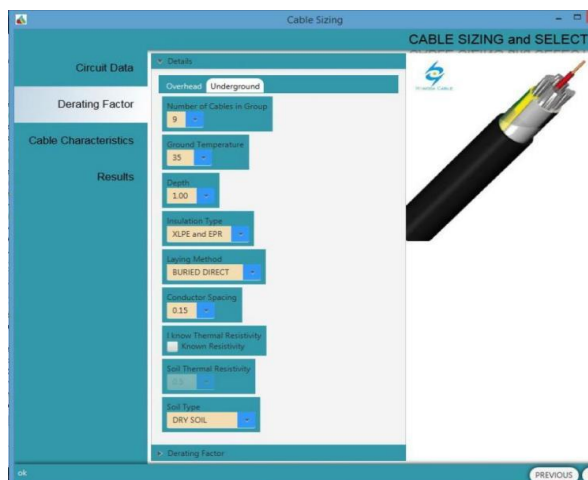


Figure 16: Derating factor interface for high voltage mains calculation

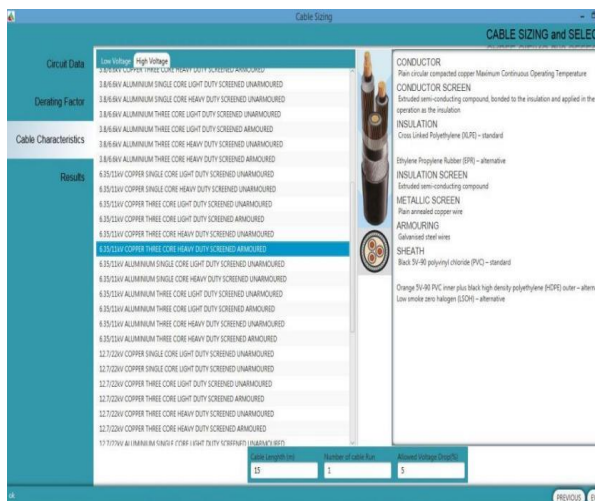


Figure 17: Cable characteristics interface for high voltage mains calculation

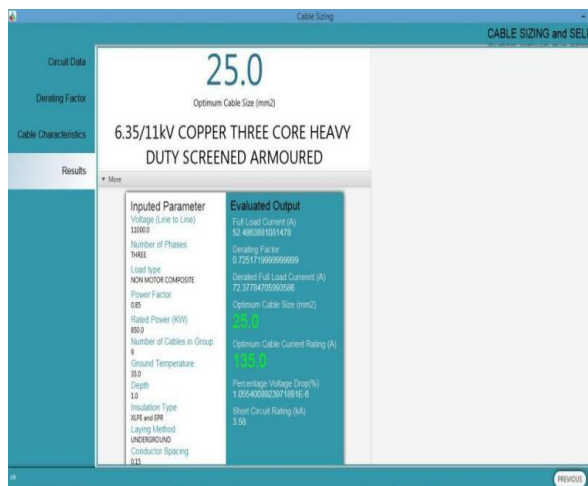


Figure 18: Result for high voltage mains calculation

#### 4. CONCLUSION

The software was found to be reasonably efficient and effective in carrying out required cable sizing calculations at computer speed and accuracy. The software serves as a design tool for the electrical engineer, student, or technician, providing an easier, faster, and more accurate means of carrying out cable sizing calculations for industrial and residential design purposes.

#### 5. CONFLICT OF INTEREST

There is no conflict of interest associated with this work.

**REFERENCES**

- BSI (2008) BS 7671: Requirements for Electrical Installations (*IET Wiring Regulations*). London: British Standards Institution.
- Coker, A. and Turner, W. (1992). *Electric Wiring: Domestic*. Oxford: Redwood Press Limited.
- Glover, J.D., Sarma, M.S. and Overbye, T.J. (2012) *Power System Analysis and Design*, 5th edn. Stamford, CT: Cengage Learning.
- Mehta, V. and Mehta, R. (2008). Principles of Power System. In Mehta, V. and Mehta, R. *Principles of Power System*. New Delhi: S.Chand, pp. 265-266.
- Nair, M. and Nithiyanthan, K. (2016). Effective Cable Sizing Model for Building Electrical Services. *Bulletin of Electrical Engineering and Informatics*, 5(1), pp. 1–7.
- Nexans Kabelmetal. (2005). *Technical Brochure for Power, Control and Instrument Signal Cable*. Lagos: Nexans Kabelmetal Nigeria.
- Olatomiwa, L. and Akinyemi, A. (2012). Design and Development of Calculator Software for Residential Electrical Services Design. *International Journal of Engineering and Technology*, 2(3) pp. 371-378.
- Open Electrical (2018) Cable Sizing Calculation. Available at: <https://openelectrical.org> (Accessed: 6 April 2025).
- Oracle Corporation (2020) Java Platform, Standard Edition Overview. Available at: <https://www.oracle.com/java/> (Accessed: 6 May 2026).
- Prysmian Cables and Systems. (2010). *Prysmian Technical Manuals (Medium Voltage)*. Liverpool: Prysmian Cables and Systems.
- Reddy, B. K. (2021). Cables and Overhead Conductors. In *Electrical Equipment*. John Wiley and Sons, pp. 329–351.
- Shelton, C. (2005). Methods of installing wiring systems, wiring enclosures and equipment. In C. Shelton, *Electrical Installations*, p. 99. Nelson Thornes Limited.
- Theraja, B.L. and Theraja, A.K. (2017) *A Textbook of Electrical Technology*. New Delhi: S. Chand Publishing.