

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL

IZAURA DE VARGAS MARTINS

DIMENSIONAMENTO AUTOMÁTICO DE VIGAS
PROTENDIDAS BIPOIADAS CONSIDERANDO A
PROTENSÃO PARCIAL

VITÓRIA

2018

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL

IZAURA DE VARGAS MARTINS

DIMENSIONAMENTO AUTOMÁTICO DE VIGAS
PROTENDIDAS BIAPOIADAS CONSIDERANDO A
PROTENSÃO PARCIAL

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Espírito Santo como requisito parcial para a obtenção do grau de Mestre em Engenharia Civil, na área de Estruturas.

Orientador: Prof. Dr. Lorenzo Augusto Ruschi e Luchi

VITÓRIA
2018

Dados Internacionais de Catalogação-na-publicação (CIP)
(Biblioteca Setorial Tecnológica,
Universidade Federal do Espírito Santo, ES, Brasil)

M386d Martins, Izaura de Vargas, 1988 -
Dimensionamento automático de vigas protendidas
biapoiadas considerando a protensão parcial / Izaura de Vargas
Martins. – 2018.
196 f. : il.

Orientador: Lorenzo Augusto Ruschi e Luchi.
Dissertação (Mestrado em Engenharia Civil) – Universidade
Federal do Espírito Santo, Centro Tecnológico.

1. Vigas de concreto protendido. 2. Software educacional.
3. Teoria das estruturas. 4. Estado limite de formação de fissuras
(ELS-F). 5. Força cortante. I. Luchi, Lorenzo Augusto Ruschi e.
II. Universidade Federal do Espírito Santo. Centro Tecnológico.
III. Título.

CDU: 624

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

DIMENSIONAMENTO AUTOMÁTICO DE VIGAS PROTENDIDAS BIAPOIADAS CONSIDERANDO A PROTENSÃO PARCIAL

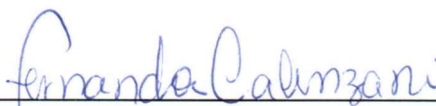
Izaura de Vargas Martins

Dissertação apresentada ao Curso de Mestrado em Engenharia Civil do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal do Espírito, como requisito parcial para obtenção do título de Mestre em Engenharia Civil, área de Estruturas.

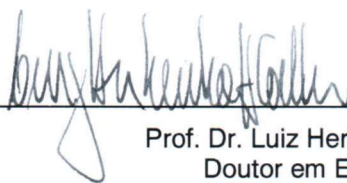
Aprovada no dia **15 de agosto de 2018** por:



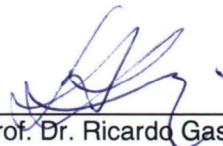
Prof. Dr. Lorenzo Augusto Ruschi e Luchi
Doutor em Engenharia Civil
Orientador – UFES



Profa Dra. Adenilcia Fernanda Grobério Calenzani
Doutora em Engenharia de Estruturas
Examinadora Interna - UFES



Prof. Dr. Luiz Herkenhoff Coelho
Doutor em Engenharia Civil
Examinador Externo – UFES



Prof. Dr. Ricardo Gaspar
Doutor em Engenharia Civil
Examinador Externo – UFABC
Por meio de video conferência

Vitória – ES, agosto de 2018

RESUMO

Este trabalho trata do estudo e elaboração de uma ferramenta computacional para o dimensionamento de vigas protendidas biapoiadas conforme a NBR 6118:2014. O programa é capaz de calcular vigas com armaduras ativas aderentes ou não aderentes, no caso de pós-tração. São contemplados os níveis de protensão completa, limitada e, especialmente, parcial. Para diferentes carregamentos, o programa realiza o pré-dimensionamento da força de protensão, calcula as perdas imediatas e progressivas. Verifica os estados-limites de serviço de compressão excessiva, descompressão, formação e abertura das fissuras, e deformações excessivas. Verifica os estados-limites últimos à flexão e à força cortante. A ferramenta foi desenvolvida utilizando o *Microsoft Excel* e o *Microsoft Visual Basic for Applications*, a partir do programa elaborado por Dominicini e Coelho (2014). Apresenta interface gráfica com o usuário e geração de memoriais de cálculo, servindo como recurso didático no meio acadêmico tanto no nível de graduação e quanto de pós-graduação. Os resultados obtidos são validados pela comparação com exemplos presentes na literatura.

Palavras-chave: vigas protendidas, protensão parcial, estado-limite de abertura de fissuras, força cortante, flecha.

ABSTRACT

This work presents a study and implementation of a computational program for the design of simply supported prestressed concrete beams according to NBR 6118: 2014. The developed software is able to design of post-tensioned prestressed concrete members with bonded or unbonded tendons. Fully, intermediary and partially prestressed concrete are contemplated. For different loads, program calculates the initial prestress force and the instantaneous and time-dependent losses. Also verifies the excessive compression, decompression limit state, the limit state of cracking and the serviceability limit of excessive deflection. Furthermore performs the ultimate limit states, flexural and shear force design. This program was developed using *Microsoft Excel* e o *Microsoft Visual Basic for Applications*, based on the software elaborated by Dominicini and Coelho (2014). It presents user graphical interface and design report, and can be used as a didactic resource in the academic environment at undergraduate and graduate levels. The results obtained are validated by comparison with examples in the literature.

Key-words: prestressed beams; partially prestressed concrete; limit state of cracking; shear force design; excessive deflection.

LISTA DE FIGURAS

Figura 2-1 – Perfil dos cabos em vigas isostáticas	24
Figura 2-2 – Perfil dos cabos em vigas hiperestáticas.....	24
Figura 2-3 – Relação de causas e efeitos nas perdas de protensão.....	25
Figura 2-4 – Comparação das perdas de protensão por atrito: ancoragens ativa/passiva e ativa/ativa.....	26
Figura 2-5 – Desenvolvimento de tensões no cabo	27
Figura 2-6 – Cabo simétrico protendido pelas duas extremidades.....	28
Figura 2-7 – Hipótese 1: $x \leq a_1$	28
Figura 2-8 – Hipótese 2: $a_1 < x < a_1 + a_2$	29
Figura 2-9 – Hipótese 3: $x = a_1 + a_2$	30
Figura 2-10 – Perdas após atrito e encunhamento – ancoragens ativa/ativa	31
Figura 2-11 – Perdas após atrito e encunhamento – ancoragens ativa/passiva	32
Figura 2-12 – Convenção adotada para o eixo y na seção transversal	46
Figura 2-13 – Domínios de estado-limite último de uma seção transversal.....	50
Figura 2-14 – Seção transversal com armadura simples	51
Figura 2-15 – Deformada no ELU.....	52
Figura 2-16 - Resultantes na seção transversal	54
Figura 2-17 – Seção transversal com armadura passiva dupla e deformada.....	55
Figura 2-18 – Equilíbrio no estágio II, seção retangular ou T com linha neutra na mesa.....	58
Figura 2-19 – Equilíbrio no estágio II, seção T com linha neutra na alma	60
Figura 2-20 – Concreto de envolvimento da armadura	63
Figura 2-21 – Redução da força cortante no caso de cabo inclinado	69
Figura 3-1 – Primeiro ícone da barra de ferramentas	74
Figura 3-2 – Propriedades geométricas da seção / concreto.....	75
Figura 3-3 – Segundo ícone da barra de ferramentas	76
Figura 3-4 – Nível de protensão / cargas.....	76
Figura 3-5 – Terceiro ícone da barra de ferramentas.....	77
Figura 3-6 – Aço / Fuso limite	77
Figura 3-7 – Quarto ícone da barra de ferramentas	78
Figura 3-8 – Geometria dos cabos e cabo equivalente	78
Figura 3-9 – Quinto ícone da barra de ferramentas	79
Figura 3-10 – Perdas imediatas e progressivas.....	79

Figura 3-11 – Sexto ícone da barra de ferramentas	80
Figura 3-12 – Estado-limite último	80
Figura 3-13 – Sétimo ícone da barra de ferramentas	80
Figura 3-14 – Verificação do estado-limite de abertura das fissuras	81
Figura 3-15 – Oitavo ícone da barra de ferramentas	82
Figura 3-16 – Dimensionamento à força cortante	82
Figura 3-17 – Distribuição dos estribos em faixas	83
Figura 3-18 – Dados para redução do cortante solicitante	83
Figura 3-19 – Janela de informação para o usuário	83
Figura 3-20 – Nono ícone da barra de ferramentas	84
Figura 3-21 – Nono ícone da barra de ferramentas	84
Figura 4-1 – Seção transversal da viga	86
Figura 4-2 – Deslocamento	89
Figura 4-3 – Seção transversal da viga	93
Figura 4-4 – Deslocamentos	95
Figura 4-5 – Seção transversal da viga	99
Figura 4-6 – Seção transversal da viga	100
Figura 4-7 – Gráfico da força ao longo do cabo (programa)	102
Figura 4-8 – Equilíbrio da seção	104
Figura 4-9 – Deslocamentos	106
Figura 4-10 – Seção transversal da viga	111
Figura 4-11 – Força ao longo do cabo (programa)	112
Figura 4-12 – Deslocamentos	116

LISTA DE QUADROS

Quadro 2.1 – Valores da fluência e da retração em função da velocidade de endurecimento do concreto	34
Quadro 2.2 – Valores numéricos usuais para a determinação da fluência e da retração.....	37
Quadro 2.3 – Valores de ψ_{1000} em porcentagem	39
Quadro 2.4 – Valores característicos superiores da deformação específica de retração $\epsilon_{cs}(t_{\infty}, t_0)$ e do coeficiente de fluência $\varphi(t_{\infty}, t_0)$	41
Quadro 2.5 – Combinações de serviço	43
Quadro 2.6 – Valores do coeficiente γ_f	43
Quadro 2.7- Classes de agressividade ambiental.....	45
Quadro 2.8 - Exigências de durabilidade relacionadas à fissuração e à proteção da armadura, em função das classes de agressividade ambiental.....	45
Quadro 2.9 – Valores mínimos para armaduras passivas aderentes.....	56
Quadro 2.10 – Taxas mínimas de armadura de flexão	57
Quadro 2.11 – Limites para deslocamento	65
Quadro 2.12 – Limites para deslocamento (continuação)	66
Quadro 2.13 – Coeficiente $\gamma_f = \gamma_{f1} \cdot \gamma_{f3}$	70

LISTA DE TABELAS

Tabela 4.1 – Protensão parcial: aplicação numérica 1.....	87
Tabela 4.2 – Verificação da flecha: aplicação numérica 1	90
Tabela 4.3 – Verificação da flecha com 5 cordoalhas CP 190 RB 12,7.....	91
Tabela 4.4 – Dimensionamento à força cortante: aplicação numérica 1	92
Tabela 4.5 – Protensão parcial: aplicação numérica 2.....	94
Tabela 4.6 – Verificação da flecha: aplicação numérica 2	97
Tabela 4.7 – Dimensionamento à força cortante: aplicação numérica 2	98
Tabela 4.8 – Protensão parcial: aplicação numérica 3.....	100
Tabela 4.9 – Resultados obtidos para o meio do vão	101
Tabela 4.10 – Perdas de protensão calculadas pelo programa	102
Tabela 4.11 – Estado-limite último (programa)	103
Tabela 4.12 – Protensão parcial: aplicação numérica 4.....	103
Tabela 4.13 – Verificação da flecha: aplicação numérica 4.....	108
Tabela 4.14 – Verificação da flecha com 5 cordoalhas CP 190 RB 15,2.....	108
Tabela 4.15 – Dimensionamento à força cortante: aplicação numérica 4.....	110
Tabela 4.16 – Resultados obtidos para o meio do vão	111
Tabela 4.17 – Perdas de protensão calculadas pelo programa	112
Tabela 4.18 – Estado-limite último (programa)	113
Tabela 4.19 – Protensão parcial: aplicação numérica 5.....	113
Tabela 4.20 – Verificação da flecha: aplicação numérica 5	118
Tabela 4.21 – Verificação da flecha com 4 cordoalhas CP 190 RB 12,7.....	118
Tabela 4.22 – Dimensionamento à força cortante: aplicação numérica 5.....	120

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Contextualização e justificativa	13
1.2	Revisão bibliográfica	14
1.3	Objetivos.....	19
1.4	Metodologia.....	20
1.5	Estrutura da dissertação	20
2	FUNDAMENTAÇÃO TEÓRICA	22
2.1	Concreto protendido	22
2.1.1	<i>Força de protensão</i>	22
2.1.2	<i>Traçado geométrico dos cabos</i>	23
2.1.3	<i>Perdas de protensão</i>	24
2.1.3.1	<i>Perdas de protensão por atrito</i>	25
2.1.3.2	<i>Perdas de protensão por deslizamento das ancoragens e acomodação das cunhas</i> ...	26
2.1.3.3	<i>Perdas de protensão por encurtamento imediato do concreto</i>	32
2.1.3.4	<i>Perdas progressivas por retração e fluência do concreto e relaxação do aço</i>	32
2.1.4	<i>Estados-limites de serviço e combinações</i>	41
2.1.5	<i>Níveis de protensão</i>	44
2.1.6	<i>Determinação da protensão</i>	46
2.1.7	<i>Estado-limite último no ato da protensão</i>	47
2.1.8	<i>Estado-limite último</i>	48
2.1.9	<i>Protensão parcial</i>	57
2.1.10	<i>Verificação do estado-limite de deformações de excessivas</i>	64
2.1.11	<i>Dimensionamento à força cortante</i>	69
3	PROGRAMA COMPUTACIONAL	74
4	APLICAÇÕES NUMÉRICAS	86
4.1	Aplicação numérica 1	86
4.1.1	<i>Protensão parcial: verificação do estado-limite de abertura das fissuras</i>	86
4.1.2	<i>Verificação do estado-limite de deformação excessiva</i>	88
4.1.3	<i>Dimensionamento à força cortante</i>	91
4.2	Aplicação numérica 2	93
4.2.1	<i>Protensão parcial: verificação do estado-limite de abertura das fissuras</i>	93
4.2.2	<i>Verificação do estado-limite de deformação excessiva</i>	94

4.2.3	<i>Dimensionamento à força cortante</i>	97
4.3	Aplicação numérica 3	99
4.3.1	<i>Protensão parcial: verificação do estado-limite de abertura das fissuras</i>	99
4.4	Aplicação numérica 4	100
4.4.1	<i>Protensão parcial: verificação do estado-limite de abertura das fissuras</i>	103
4.4.2	<i>Verificação do estado-limite de deformação excessiva</i>	105
4.4.3	<i>Dimensionamento à força cortante</i>	109
4.5	Aplicação numérica 5	110
4.5.1	<i>Protensão parcial: verificação do estado-limite de abertura das fissuras</i>	113
4.5.2	<i>Verificação do estado-limite de deformação excessiva</i>	115
4.5.3	<i>Dimensionamento à força cortante</i>	118
5	CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS	121
5.1	Conclusões	121
5.2	Sugestões para trabalhos futuros	122
	REFERÊNCIAS	124
	APÊNDICE 1	129

1 INTRODUÇÃO

1.1 Contextualização e justificativa

A protensão consiste em aplicar esforços que visam limitar ou, até mesmo, eliminar as tensões de tração no concreto, reduzir as fissuras, controlar os deslocamentos, aumentar a resistência, diminuir a quantidade de armadura passiva e reduzir o peso da estrutura. Portanto, esta técnica ganha, cada vez mais, espaço como solução estrutural de diversas construções, sendo grandes ou pequenas, como em pontes, viadutos, galpões pré-moldados e edifícios.

A contribuição da protensão para a engenharia atual é inestimável, tornando possível a execução de estruturas antes inviáveis ao concreto armado convencional. No entanto, como qualquer tecnologia, se mal utilizada, seja por calculistas, empresas de protensão ou construtoras, na tentativa de reduzir custos, de acelerar a produção ou mesmo por falha técnica, pode gerar danos às edificações, causando fissuração e deslocamentos excessivos e até o colapso da estrutura.

Desse modo, é necessário investir no aperfeiçoamento dos métodos de execução, cálculo e dimensionamento de peças protendidas. Além disso, as exigências de produção rápida e eficiente do mercado da construção civil tornam fundamental a implementação de rotinas computacionais para a realização das análises estruturais e dimensionamento de elementos protendidos.

A NBR 6118:2014 determina três níveis de protensão: completa, limitada e parcial. A completa não admite tensões de tração na seção transversal, a limitada permite tensões de tração desde que não ultrapasse a tensão resistente à tração do concreto. A protensão parcial é o nível menos intenso de protensão, que permite tensões de tração, desde que seja verificado o estado-limite de abertura das fissuras.

A protensão parcial possibilita a dosagem racional entre as armaduras passivas e ativas, como comenta Emerick (2005). Assim, por proporcionar estruturas econômicas, vem sendo largamente utilizada nos projetos de lajes planas de edifícios residenciais e comerciais.

Entretanto, a protensão parcial implica numa análise mais complexa, por se tratar de estruturas no estágio II, o que dificulta o dimensionamento, sendo necessária a verificação do estado-limite de abertura de fissuras e verificação do estado-limite de deformação excessiva considerando seção fissurada. Soma-se ao fato de que a literatura é limitada no estudo da protensão parcial e pouco apresenta sobre o cálculo de flechas em peças protendidas fissuradas.

Além disso, são poucos os programas que realizam o dimensionamento de estruturas no nível de protensão parcial, seguindo a ABNT NBR 6118:2014 – Projeto de Estruturas de Concreto – Procedimento, e os que existem, geralmente são comerciais, podendo-se citar o TQS como um dos principais. Com isso, a elaboração de ferramentas computacionais de livre disponibilização na área de dimensionamento de vigas protendidas se apresenta como um campo profícuo de pesquisa.

Atualmente, é inimaginável o desenvolvimento de projetos estruturais sem o auxílio de programas computacionais. Além disso, programas com interface gráfica e geração de memoriais de cálculo fornecem, aos estudantes de engenharia, boa visualização das etapas de cálculos realizados, servindo como recurso didático no meio acadêmico, tanto no nível de graduação, quanto de pós-graduação. Assim, neste trabalho foi desenvolvido um programa que contemple o dimensionamento de vigas parcialmente protendidas.

1.2 Revisão bibliográfica

O concreto é um material frágil e, mesmo quando submetido a tensões de tração relativamente baixas, pode apresentar fissuração. Leonhardt (1983) explica que a deficiente resistência à tração do concreto fez com que, desde o início, se pensasse em colocar sob compressão a área tracionada. Vela et al. (1993) complementam dizendo que o concreto protendido como técnica estrutural teve sua origem no desejo de resolver o problema da fissuração. Pfeil (1983) explica que a protensão consiste em introduzir esforços capazes de anular ou minimizar as tensões de tração no concreto e a abertura de fissuras.

Segundo Aalami (2014), um dos pioneiros na protensão foi o engenheiro P. H. Jackson que, no ano de 1872, em São Francisco – EUA, patenteou um sistema de pós-tensão que consistia na inserção de barras de aço em blocos cerâmicos que seriam tensionados por um dispositivo

roscado. Em 1888, C. W. Doebling patenteou a um sistema de protensão de lajes com fios metálicos na Alemanha. Entretanto com as propriedades do aço disponível na época e altas perdas de protensão, as tentativas iniciais não tiveram sucesso.

Aalami (2014) esclarece que a partir do desenvolvimento do Método do Balanceamento das Cargas por T. Y. Lin (1963), a utilização das lajes protendidas aumentou muito nos Estados Unidos. O método consiste no cálculo da força de protensão necessária e traçado dos cabos em elevação para equilibrar certa parcela do carregamento externo atuante. Aalami (2007) explica que em 1963, o concreto protendido foi incluído à norma American Concrete Institute - ACI 318. Com a expansão da protensão para outros países, outras normas também incluíram o sistema.

O desenvolvimento de ferramentas computacionais para o dimensionamento de estruturas de concreto armado e protendido, assim como para o ensino de engenharia, é um campo de fundamental importância para a pesquisa, uma vez que a utilização de muitos programas fica limitada ao escritório técnico desenvolvedor.

Dessa forma, ferramentas de auxílio ao ensino e prática de engenharia vêm sendo difundidas, Ventri e Lindenberg Neto (2001) apontam a importância da utilização da informática e o desenvolvimento de animações tanto no ensino de conceitos fundamentais da mecânica das estruturas, como na obtenção das equações diferenciais de equilíbrio e obtenção de esforços solicitantes. Assis et al. (2002) descrevem o desenvolvimento de material didático multimídia via web para as disciplinas de concreto armado e concreto protendido, obtendo bons resultados de aprendizagem por parte dos alunos. O material e os programas elaborados podem ser obtidos no site do laboratório de mecânica computacional da Escola Politécnica da Universidade de São Paulo.

Lindquist (2002) elabora um programa computacional para o dimensionamento e detalhamento de pórticos planos de concreto armado com fins didáticos para disciplinas de concreto armado. O programa funciona como um pós-processador, obtém dados como esforços e deslocamentos nodais de um pré-processador existente, e realiza o dimensionamento e detalhamento das peças. Seus resultados são comparados com o *DTool* e com o TQS.

Nóbrega e Horowitz (2003) desenvolvem um programa para a determinação do traçado de cabos em vigas contínuas protendidas. A ferramenta é de fácil utilização e permite a visualização dos perfis que são compostos por segmentos concordantes “parabólico – retilíneo – parabólico”. Definido o traçado, são calculadas as perdas imediatas de protensão, em seguida os hiperestáticos de protensão e depois as perdas progressivas. Então, são calculadas as tensões nas fibras extremas da seção transversal para as diversas etapas de protensão. O programa realiza a otimização dos cabos com as variáveis força de protensão e elevação do cabo resultante. Entretanto, a ferramenta não dimensiona as armaduras passivas.

Mello (2005) apresenta processos de cálculo da armadura ativa para lajes lisas e possibilidades de variação na adoção de quantitativos de armadura ativa e passiva. Discute possíveis perfis de cabos, distribuições em planta e uso de cordoalhas aderentes ou não. A análise é realizada conforme a NBR 6118:2003 e utiliza o programa TQS.

Jost (2006) implementa um modelo não linear físico e geométrico utilizando o método dos elementos finitos para análise de estruturas de concreto armado e protendido com ou sem aderência. Utiliza a linguagem de programação FORTRAN 90/95 e elabora uma interface gráfica. Os resultados obtidos no programa são comparados com exemplos da literatura, e são considerados satisfatórios.

Milani (2006) apresenta um modelo numérico, utilizando elementos finitos, para o dimensionamento de lajes lisas protendidas. Foi utilizada a linguagem de programação FORTRAN 90. Propõe um modelo para determinação da elevação do cabo e perdas imediatas de protensão. Analisa as tensões de serviço e tensões últimas. Entretanto, o programa não realiza o cálculo das perdas de protensão progressivas e se limita a utilização de cordoalhas engraxadas.

Klein e Loriggio (2006) utilizam a linguagem de programação *Visual Basic*, a mesma proposta para este trabalho, para desenvolver uma rotina computacional para o dimensionamento de vigas simplesmente apoiadas de concreto protendido, verificando-se os estados-limite de serviço e considerando-se pós-tração. Os resultados obtidos são satisfatórios e o autor indica que o programa pode ser utilizado tanto para o ensino quanto para profissionais do ramo. O programa não dimensiona vigas no nível de protensão parcial e também não verifica o estado-limite último.

Hernandez et al. (2007) apresentam um programa para a otimização da distribuição dos cabos de vigas e lajes pré-fabricadas protendidas, que pode ser utilizado por projetistas de pontes. A ferramenta analisa elementos com até três cabos e duas classes de perdas de protensão, considera momentos fletores e forças normais na seção e apresenta uma interface gráfica de fácil utilização. Os resultados obtidos são considerados satisfatórios.

Seguindo os moldes de programas educacionais, Sales (2010) elabora um programa, via web, utilizando *html* e *Java*, para o dimensionamento de seções retangulares de concreto armado, submetidas à flexão normal simples, à flexão normal composta e ao cisalhamento, em que o usuário pode visualizar o esboço do detalhamento da seção por diferentes ângulos em 3D.

Em Lazzari et al. (2013) é proposta uma ferramenta automática para o cálculo de vigas de concreto submetidas à flexão, considerando protensão aderente e não aderente. A análise estrutural utiliza elementos finitos considerando a não linearidade geométrica. O programa permite analisar fases construtivas e carregamento cíclico. São levadas em consideração os níveis de protensão completa, limitada e parcial. Os cálculos são baseados na norma brasileira NBR 6118:2007 e na francesa Règles BPEL 91, obtendo-se resultados semelhantes.

Dominicini e Coelho (2014) desenvolvem um programa utilizando o *Microsoft Excel* e o *Visual Basic Applications* para o dimensionamento de vigas biapoiadas protendidas, aderentes ou não, com pré-tensão ou pós-tração, para os níveis de protensão completa e limitada, de acordo com a NBR 6118:2007. É realizada a validação do programa por meio de comparações com exemplos presentes na literatura. Porém, o software não dimensiona à força cortante, não abrange o nível de protensão parcial e não verifica o estado-limite de deformação excessiva.

Em Nacht (2015) é desenvolvida uma ferramenta computacional, chamada *Prestress*, para a verificação ao estado-limite de serviço de vigas de concreto protendido com pós-tração aderente. O programa é uma extensão para Autodesk Robot Structural Analysis Professional e apresenta duas saídas: uma planilha contendo os esforços e as tensões atuantes na viga, ao longo de sua vida-útil, e a verificação dessas tensões em relação ao estado-limite de serviço e o Robot apresenta os carregamentos equivalentes da protensão. No entanto, o *software* não realiza a análise para a protensão parcial e não verifica o estado-limite último.

Nacht (2015) ainda apresenta um resumo das principais ferramentas do mercado para o cálculo de vigas protendidas, como segue:

- Robot: O programa inclui um carregamento equivalente de protensão, com apenas uma curvatura e desconsidera as perdas.
- SAP 2000: O programa calcula as perdas por normas internacionais e não apresenta verificações de estados-limite de serviço. Permite o cálculo de pré-tração e pós-tração, sendo que os cabos podem ser inseridos apenas como objetos ou carregamento equivalente.
- ADAPT-PT/RC: calcula vigas com ou sem aderência, pré e pós-tensionadas, no estado-limite de serviço, conforme diferentes normas, inclusive a NBR 6118.
- Nemetschek Scia: Também verifica vigas protendidas em 3D, com e sem aderência, pré e pós-tensionadas. Calcula as perdas e verifica os estados-limites de serviço e último, mas não possui a NBR 6118 em sua biblioteca.

Inclui-se também:

- TQS: Programa brasileiro que calcula e detalha vigas e lajes protendidas de acordo com a NBR 6118:2014, calcula as perdas de protensão e verifica os estados-limites de serviço e último.

Silva (2015) apresenta o desenvolvimento de um programa para a verificação de seções poligonais de concreto armado e protendido submetidas à flexão composta oblíqua de acordo com a NBR 6118:2014, a ferramenta também calcula a relação momento – normal – curvatura que pode ser utilizada para outras análises como, por exemplo, o método geral para o cálculo de pilares. As rotinas utilizam a linguagem *Pascal* com o compilador *Lazarus*. Salienta-se que o programa verifica as seções protendidas a partir do fornecimento pelo usuário dos dados dos materiais, da geometria e das armações passivas e ativas. Primeiro é analisado o estado-limite último e depois o estado-limite de serviço.

Monteiro et al. (2016) implementam um programa computacional para o dimensionamento das armaduras longitudinais de uma seção de um elemento protendido, submetido à flexão normal, seguindo a NBR 6118:2014 e utilizando o *software* MATLAB. Verifica os estados-limites de serviço e último, considera protensão aderente e não aderente, protensão completa, limitada e parcial. Não calcula os momentos solicitantes e nem as perdas de protensão, ambos são dados de entrada. A ferramenta calcula seções transversais individuais, a partir da

geometria e alturas úteis fornecidas pelo usuário. Para validação da ferramenta desenvolvida, seus resultados são comparados com exemplos de vigas na literatura.

Labadan (2016) desenvolve um programa utilizando o *Microsoft Excel* e o *Visual Basic Applications* para o dimensionamento de vigas contínuas protendidas pós-tracionadas, visando à simplificação do processo de tentativa e erro do cálculo de estruturas protendidas. A planilha desenvolvida expõe os cálculos envolvidos no dimensionamento e permite a realização de modificações nos parâmetros de projeto e análise rápida de suas influências nos resultados. A verificação das tensões máximas admissíveis pode ser realizada de acordo com o ACI, AASHTO ou PCI Code.

Deve-se destacar que o Programa de Pós-graduação da Universidade Federal do Espírito Santo (Ufes) tem atuado na pesquisa de elementos protendidos, podendo-se citar, dentre outros, Leite Júnior (2015), que estudou o fenômeno da punção em lajes protendidas com cordoalhas não aderentes, Carneiro (2015) que analisou modelos de dimensionamento à flexão destas lajes e Ferreira (2013) que estudou o desempenho e critérios de abordagem para lajes lisas protendidas.

A partir da revisão bibliográfica, observa-se que o concreto protendido é amplamente estudado, tanto no que diz respeito a métodos de análise quanto ao desenvolvimento de ferramentas computacionais para dimensionamento e verificação. Entretanto, poucos são os programas que abordam vigas no nível de protensão parcial, e a verificação dos estados-limites de serviço de abertura das fissuras e deformação excessiva, evidenciando-se a necessidade de mais estudos nesta área.

1.3 Objetivos

Este trabalho visa ao estudo e desenvolvimento de equações para determinação da tensão nas armaduras passivas de peças submetidas à protensão parcial, além da elaboração de uma ferramenta computacional para o dimensionamento de vigas protendidas biapoiadas, parcialmente protendidas, que possa ser utilizada no meio acadêmico como recurso didático adicional na disciplina Concreto Protendido.

1.4 Metodologia

O programa foi desenvolvido a partir da ferramenta elaborada em Dominicini e Coelho (2014). É capaz de calcular vigas protendidas biapoiadas com armaduras ativas aderentes ou não aderentes, no caso de pós-tração, para concretos até a classe C50, conforme a NBR 6118:2014.

Antes da implementação do código do programa, foi necessária a realização da revisão bibliográfica, e desenvolvimento das equações para determinação da tensão nas armaduras passivas de peças submetidas à protensão parcial, assunto pouco abordado na literatura.

O programa realiza as seguintes etapas de cálculo:

- Para cada nível de protensão, determina a força de protensão necessária, o alojamento e o traçado dos cabos;
- calcula as perdas de protensão imediatas e progressivas;
- verifica os estados-limites de serviço;
- dimensiona as armaduras passivas de flexão para o estado-limite último;
- verifica os estados-limites de abertura das fissuras e deformação excessiva;
- dimensiona à força cortante.

Com a intenção de abranger grande número de usuários, o programa foi desenvolvido com interface gráfica e geração de memoriais de cálculo. Utilizou-se o *Microsoft Excel* e o *Microsoft Visual Basic for Applications*, uma vez que a maioria dos alunos e profissionais tem acesso a estes *softwares* e facilidade em sua utilização.

1.5 Estrutura da dissertação

A dissertação é composta por cinco capítulos. No primeiro capítulo é apresentada a contextualização do tema, a exposição da importância da protensão para a engenharia civil e arquitetura atual e a justificativa do tema proposto para a pesquisa, indicando as necessidades do campo em estudo.

No segundo capítulo são apontadas as considerações de cálculo de estruturas pretendidas e todas as etapas necessárias ao seu dimensionamento. O capítulo contempla os cálculos necessários ao programa computacional.

O terceiro capítulo traz a apresentação do programa desenvolvido, e expõe suas janelas, desde a tela inicial e os dados de entrada do problema, até os resultados finais obtidos.

No capítulo quatro são realizados testes e verificações dos resultados encontrados pelo programa, que são comparados com exemplos da literatura.

O último capítulo, enfim, apresenta as conclusões e as considerações finais. Nele são discutidos os resultados encontrados no trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Concreto protendido

A baixa resistência do concreto à tração fez com que se pensasse, inicialmente, em submeter à compressão as zonas tracionadas das peças de concreto. No princípio da protensão, a ideia era eliminar completamente as tensões de tração. Já atualmente, permite-se haver tração e fissuração limitada nos elementos. Naaman (2012) explica que a protensão consiste na aplicação de um estado prévio de tensões, visando a melhorar o comportamento da estrutura. A ABNT NBR 6118:2014 Projeto de estruturas de concreto – Procedimento define, no item 3.1.4, elementos de concreto protendido como:

Aqueles nos quais parte das armaduras é previamente alongada por equipamentos especiais de protensão, com a finalidade de, em condições de serviço, impedir ou limitar a fissuração e os deslocamentos da estrutura, bem como propiciar o melhor aproveitamento de aços de alta resistência no estado-limite último (ELU).

Ainda segundo a NBR 6118:2014, os materiais empregados são o concreto de classe C25 ou superior, as armaduras passivas CA-25, CA-50 e CA-60 e as armaduras ativas, barras de aço-liga, fios e cordoalhas de relaxação normal ou baixa, conforme as indicações da ABNT NBR 7482:2008, fios de aço para estruturas de concreto protendido – especificação e ABNT NBR 7483:2008 cordoalhas de aço para estruturas de concreto protendido – especificação.

A protensão pode ser classificada quanto ao sistema de aplicação e aderência. Na pré-tração, os cabos são tracionados antes da concretagem e a aderência entre a armadura ativa e o concreto se inicia em seu lançamento. Este sistema é utilizado na pré-fabricação de elementos e apresenta traçado retilíneo e poligonal dos cabos. Na pós-tração, a protensão é aplicada após o endurecimento do concreto, permitindo, portanto, o traçado parabólico dos cabos. Este sistema pode ser aderente: em que a aderência entre a armadura ativa e o concreto se dá por meio da injeção de calda de cimento nas bainhas dos cabos após a protensão; ou sem aderência: com a utilização de cordoalhas engraxadas.

2.1.1 Força de protensão

De acordo com a NBR 6118:2014, a força média na armadura de protensão na abscissa x e no tempo t é dada pela equação (2.1)

$$P_t(x) = P_0(x) - \Delta P_t(x) = P_i - \Delta P_0(x) - \Delta P_t(x) \quad (2.1)$$

em que

$P_t(x)$ é força na armadura de protensão no tempo t , na seção da abscissa x ;

$P_0(x)$ é força na armadura de protensão no tempo $t = 0$, na seção da abscissa x ;

$\Delta P_t(x)$ é a perda de protensão no tempo t , calculada após o tempo $t = 0$, na seção da abscissa x ;

P_i é a força máxima aplicada à armadura de protensão pelo equipamento de tração;

$\Delta P_0(x)$ é a perda imediata de protensão, medida a partir de P_i no tempo $t = 0$, na seção da abscissa x .

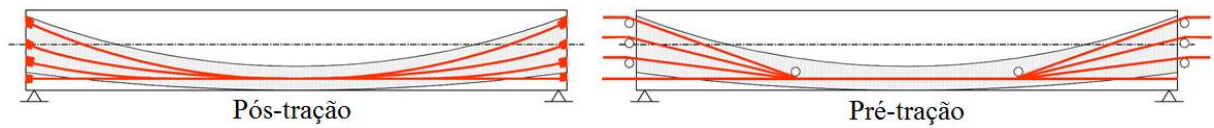
A força de tração nas armaduras durante as operações de protensão é limitada por valores de tensões no aço. Na pós-tração, devem-se respeitar os limites de $0,74 f_{ptk}$ e $0,87 f_{pyk}$ para aços de relaxação normal e $0,74 f_{ptk}$ e $0,82 f_{pyk}$ para relaxação baixa. No caso de cordoalhas engraxadas de relaxação baixa, permite-se aumentar as tensões limites para $0,80 f_{ptk}$ e $0,88 f_{pyk}$. E para barras de aço CP-85/105 os limites são $0,72 f_{ptk}$ e $0,88 f_{pyk}$. Após o término da protensão, as verificações de segurança devem ser realizadas de acordo com os estados-limites.

2.1.2 Traçado geométrico dos cabos

Conforme Veríssimo e César (1998), o traçado dos cabos é muito importante na configuração final dos esforços nos elementos protendidos. Sempre que possível, o perfil dos cabos deve acompanhar os diagramas de momentos fletores solicitantes, para que os esforços devidos à protensão variem proporcionalmente aos devidos aos carregamentos externos.

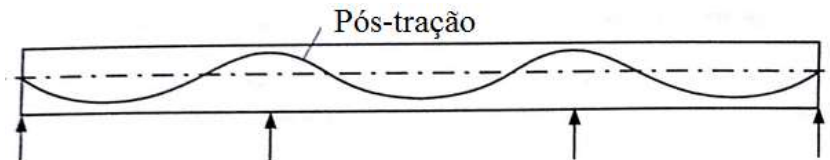
A resultante das componentes normais dos cabos deve estar no núcleo central de inércia da seção de ancoragem. Então, devem ser realizadas as adaptações de acordo com a exigência construtiva, respeitados os cobrimentos, espaçamentos máximos e mínimos, dimensões dos nichos e placas de ancoragem. A seguir, a Figura 2-1 mostra o traçado de cabos em vigas isostáticas e a Figura 2-2 mostra o perfil de uma viga hiperestática.

Figura 2-1 – Perfil dos cabos em vigas isostáticas



Fonte: Hanai (2005).

Figura 2-2 – Perfil dos cabos em vigas hiperestáticas



Fonte: Adaptada de Naaman (2012).

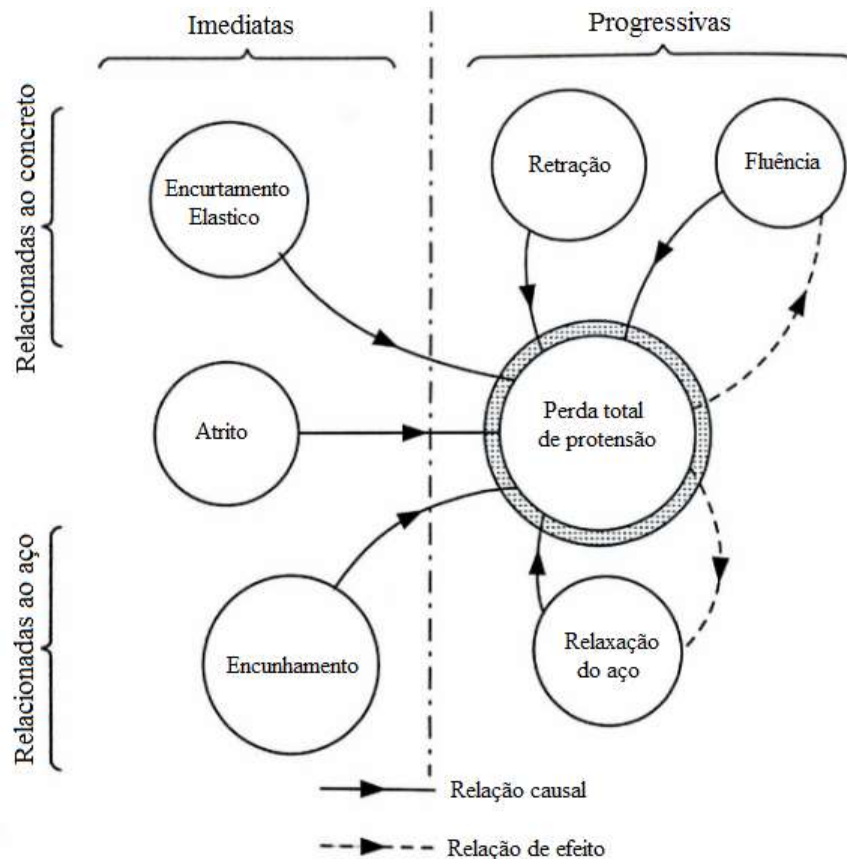
2.1.3 Perdas de protensão

De acordo com a NBR 6118:2014, devem ser previstas as perdas de protensão em relação à força inicial aplicada, ocorridas durante a transferência (imediatas) e ao longo do tempo (progressivas).

Na pós-tração, as perdas imediatas são consequência do atrito entre o aço e o concreto ou bainhas, do deslizamento das armaduras na ancoragem e acomodação das cunhas, e do encurtamento elástico do concreto.

São progressivas as perdas por retração e fluência do concreto e por relaxação do aço. Segundo Nawy (2010), as forças de protensão aplicadas nas estruturas sofrem redução em sua intensidade durante um período de aproximadamente cinco anos. Naaman (2012) ilustra as perdas de protensão no esquema da Figura 2-3.

Figura 2-3 – Relação de causas e efeitos nas perdas de protensão



Fonte: Adaptada de Naaman (2012)

2.1.3.1 Perdas de protensão por atrito

As perdas por atrito entre o cabo e a bainha ou concreto ocorrem no sistema de pós-tração e aumentam nos trechos curvos, por conta das elevadas pressões de contato nos desvios do traçado dos cabos. Essas perdas são calculadas pela equação (2.2), conforme a NBR 6118:2014:

$$\Delta P(x) = P_i [1 - e^{-(\mu \sum \alpha + kx)}] \quad (2.2)$$

em que

P_i é a força máxima aplicada à armadura de protensão pelo equipamento de tração;

x é a abscissa, em metros, do ponto onde se calcula ΔP , medida a partir da ancoragem;

$\sum \alpha$ é o somatório dos ângulos, em radianos, de desvio entre a ancoragem e o ponto x ;

μ é o coeficiente de atrito entre o cabo e a bainha, valor em 1/radianos, que na falta de dados experimentais, devem ser adotados os valores apresentados a seguir:

$\mu = 0$ entre cabo e concreto;

$\mu = 0,3$ entre barras ou fios com mossas ou saliências e bainha metálica;

$\mu = 0,2$ entre fios lisos ou cordoalhas e bainha metálica;

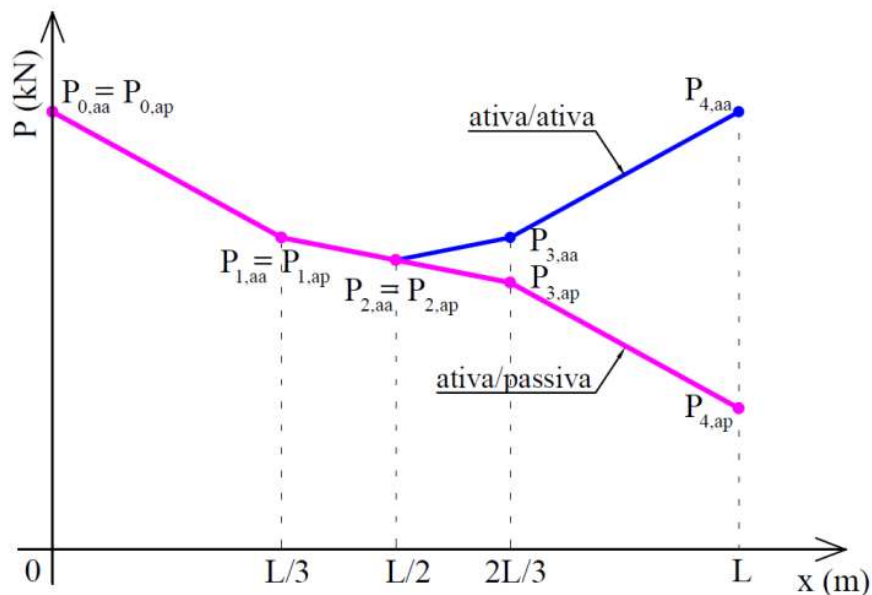
$\mu = 0,1$ entre fios lisos ou cordoalhas e bainha metálica lubrificada;

$\mu = 0,05$ entre cordoalha e bainha de polipropileno lubrificada;

k é o coeficiente de perda por metro causada por curvas não intencionais do cabo, que na falta de valores experimentais, deve-se usar $k = 0,01 \mu$ (1/m).

A Figura 2-4 ilustra a comparação das perdas por atrito de uma viga protendida com ancoragens ativa/passiva e ancoragens ativa/ativa.

Figura 2-4 – Comparação das perdas de protensão por atrito: ancoragens ativa/passiva e ativa/ativa



Fonte: Adaptada de Luchi (2016).

2.1.3.2 Perdas de protensão por deslizamento das ancoragens e acomodação das cunhas

As perdas de protensão por acomodação das ancoragens ocorrem na pós-tração, em que há encunhamento individual das cordoalhas, e dependem do tipo de ancoragem e sistema de protensão. Na transferência da protensão ocorre o deslizamento da cordoalha na ancoragem, retornando para a bainha, o que provoca perdas em determinado trecho dos cabos.

Carvalho (2012) apresenta o desenvolvimento de tensões (1-2) num cabo antes de ser ancorado (ver Figura 2-5). Após a ancoragem, obtém-se o desenvolvimento (3-4-2). Próximo

à cunha, o decréscimo de tensão vale $\Delta\sigma$ e vai reduzindo até chegar a zero no ponto 4. A queda de tensão diminui porque o atrito entre o cabo e a bainha impede o movimento do cabo retornando para dentro da bainha. Os pontos no trecho (4-2) não se movimentam durante o processo de ancoragem e, por isso, não sofrem decréscimo de tensão.

Ao analisar um trecho dx , na Figura 2-5, tem-se as expressões (2.3) e (2.4)

$$\sigma = E \cdot \varepsilon \quad (2.3)$$

$$\varepsilon = \frac{\Delta(dx)}{dx} \quad (2.4)$$

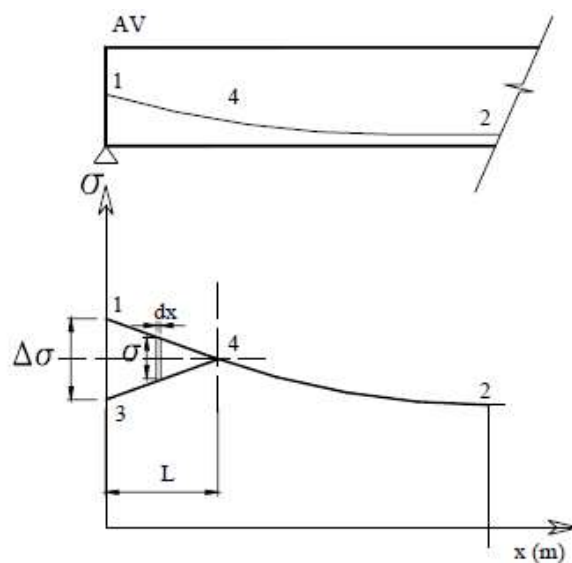
em que $\Delta(dx)$ é o encurtamento do trecho do cabo e σ é a perda de tensão devido a acomodação da ancoragem. Logo, têm-se (2.5) e (2.6)

$$\sigma = E \cdot \frac{\Delta(dx)}{dx} \quad (2.5)$$

$$\Delta l = \int_0^L \Delta(dx) = \frac{1}{E} \int_0^L \sigma \cdot dx \quad (2.6)$$

Na expressão (2.6), Δl é o encurtamento total do cabo e $\int_0^L \sigma \cdot dx$ é a área compreendida entre (1-3-4) na Figura 2-5. Os valores de encurtamento são fornecidos pelos fabricantes das ancoragens ou sistema de protensão e podem ser determinados experimentalmente. Por hipótese, o coeficiente de atrito é o mesmo tanto em carga como em descarga.

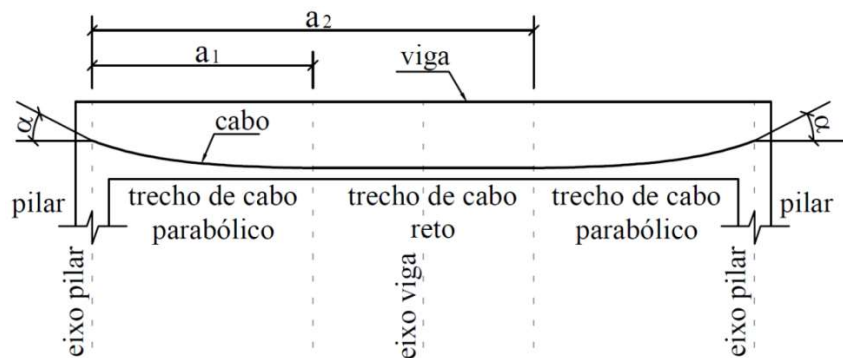
Figura 2-5 – Desenvolvimento de tensões no cabo



Fonte: Carvalho (2012).

Conforme Ishitani (1998), seja o cabo simétrico da Figura 2-6 protendido pelas duas extremidades, há três hipóteses para o alcance (x) das perdas de protensão devido à acomodação das ancoragens: primeira hipótese, o encunhamento afeta apenas o trecho curvo do cabo; segunda hipótese, alcança parte do trecho reto, e na terceira, as perdas influenciam todo o comprimento da peça.

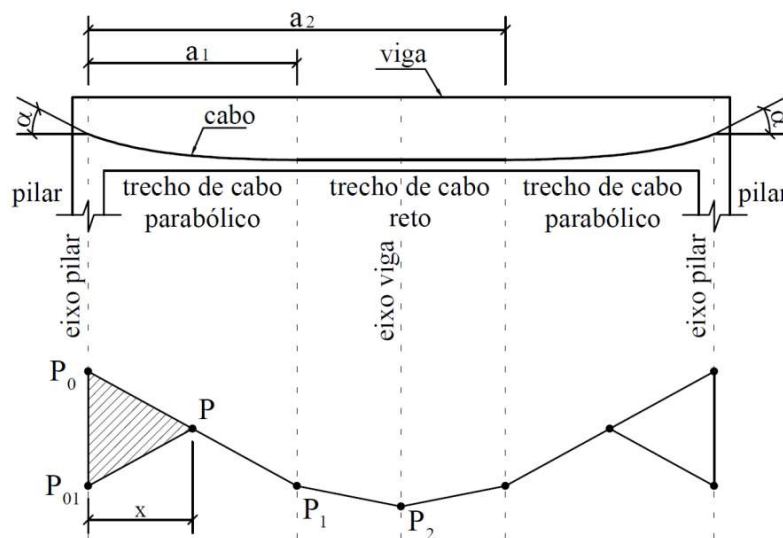
Figura 2-6 – Cabo simétrico protendido pelas duas extremidades



Fonte: Adaptada de Ishitani (1998).

A Figura 2-7 refere-se à primeira hipótese, caso $x \leq a_1$, na qual o encunhamento afeta somente o trecho parabólico.

Figura 2-7 – Hipótese 1: $x \leq a_1$



Fonte: Adaptada de Ishitani (1998).

Ao observar a área hachurada da Figura 2-7, a equação (2.7)

$$\Delta l = \frac{2(P_0 - P)x}{2E_p A_p} = \left(P_0 - P_1 \frac{x}{a_1} \right) \cdot \frac{x}{E_p \cdot A_p} \quad (2.7)$$

em que E_p e A_p são o módulo de elasticidade e área da seção da armadura ativa, respectivamente, e Δl é o alongamento do cabo no ato da protensão. Das perdas de atrito, tem-se (2.8)

$$P_1 = P_0 \cdot e^{-(\mu\alpha + ka_1)} \cong P_0(1 - \mu\alpha - ka_1) \quad (2.8)$$

Substituindo a equação (2.8) em (2.7), chega-se expressão (2.9)

$$\Delta l = \left(P_0 - P_0(1 - \mu\alpha - ka_1) \frac{x}{a_1} \right) \cdot \frac{x}{E_p \cdot A_p} = P_0 \frac{(\mu\alpha + ka_1)x}{a_1} \cdot \frac{x}{E_p \cdot A_p} \quad (2.9)$$

Ao isolar x , obtém-se (2.10)

$$x = \sqrt{\frac{E_p \cdot A_p \cdot \Delta l \cdot a_1}{P_0(\mu\alpha + ka_1)}} \leq a_1 \quad (2.10)$$

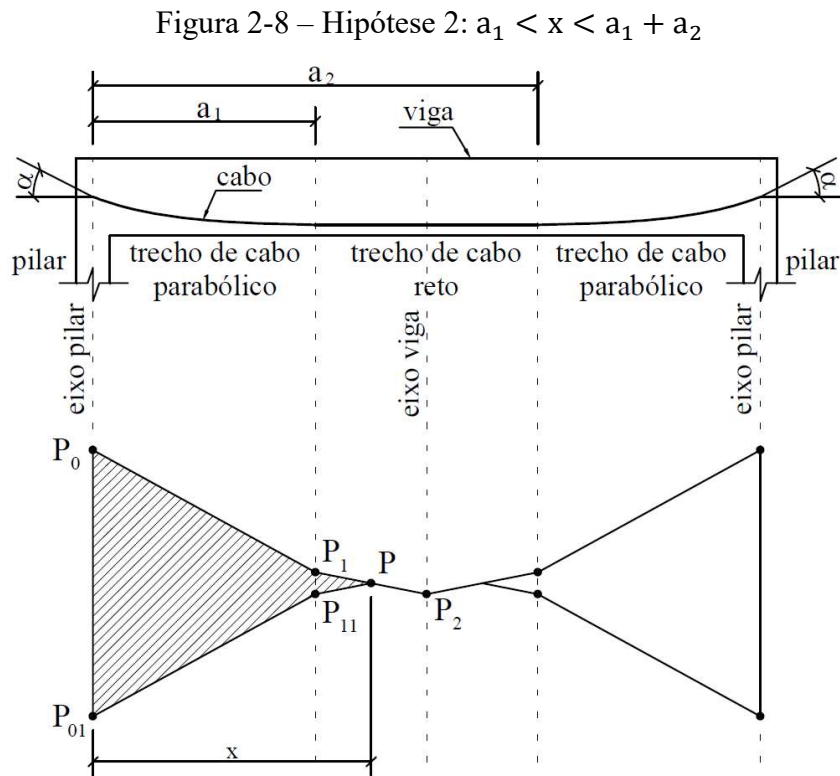
Pode-se, assim, encontrar o valor de P ao substituir (2.10) na equação (2.11)

$$P = P_0 \left(1 - \mu\alpha \frac{x}{a_1} - kx \right) \quad (2.11)$$

Finalmente, P_{01} é dado por (2.12)

$$P_{01} = 2P - P_0 \quad (2.12)$$

Para a segunda hipótese, $a_1 < x < a_1 + a_2$, na qual o encunhamento afeta o trecho reto do cabo (ver Figura 2-8).



Fonte: Adaptada de Ishitani (1998).

Pela área hachurada da Figura 2-8, chega-se a equação (2.13)

$$\Delta l = \frac{1}{E_p \cdot A_p} \cdot \{[(P_0 - P) + (P_1 - P)] \cdot a_1 + (P_1 - P) \cdot (x - a_1)\} \quad (2.13)$$

Ao substituir (2.14) e (2.15) em (2.13) e ao simplificar, encontra-se a expressão (2.16)

$$P_1 \cong P_0(1 - \mu\alpha - ka_1) \quad (2.14)$$

$$P \cong P_1[1 - k(x - a_1)] \quad (2.15)$$

$$\Delta l = \frac{1}{E_p \cdot A_p} \cdot \left[P_0(\mu\alpha + ka_1)a_1 + 2P_1k(x - a_1) \left(a_1 + \frac{x - a_1}{2} \right) \right] \quad (2.16)$$

Então, é possível isolar x e obter (2.17)

$$a_1 < x = \sqrt{\frac{E_p A_p \Delta l - (P_0 - P_1)a_1 + P_1 k a_1^2}{P_1 k}} < a_1 + a_2 \quad (2.17)$$

Pode-se, assim, encontrar o valor de P ao substituir (2.17) na equação (2.18)

$$P = P_1[1 - k(x - a_1)] \quad (2.18)$$

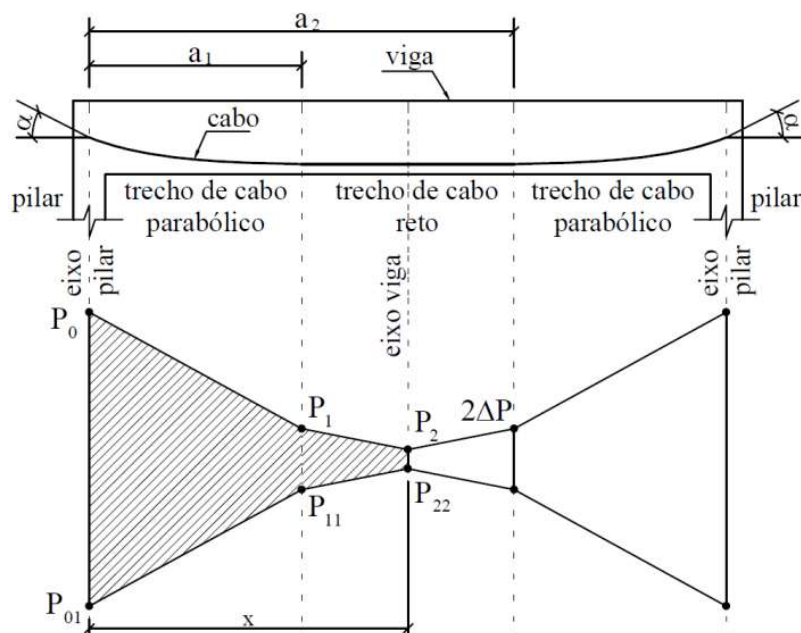
Finalmente, P_{01} e P_{11} são dados pelas equações (2.19) e (2.20), respectivamente

$$P_{01} = 2P - P_0 \quad (2.19)$$

$$P_{11} = 2P - P_1 \quad (2.20)$$

Para a terceira hipótese, $x = a_1 + a_2$, em que o encunhamento afeta todo o cabo (ver Figura 2-9). Observa-se pela área hachurada que Δl pode ser dado pela equação (2.21).

Figura 2-9 – Hipótese 3: $x = a_1 + a_2$



Fonte: Adaptada de Ishitani (1998).

$$\Delta l = \frac{1}{E_p \cdot A_p} \left[(P_0 - P_1) a_1 + 2(P_1 - P_2) \left(a_1 + \frac{a_2}{2} \right) + 2\Delta P (a_1 + a_2) \right] \quad (2.21)$$

Ao isolar ΔP em (2.21), obtém-se a expressão (2.22)

$$\Delta P = \frac{E_p A_p \frac{\Delta l}{2} - \frac{P_0 - P_1}{2} a_1 - (P_1 - P_2) \left(a_1 + \frac{a_2}{2} \right)}{a_1 + a_2} \quad (2.22)$$

Finalmente, são dados pelas equações (2.23), (2.24) e (2.25)

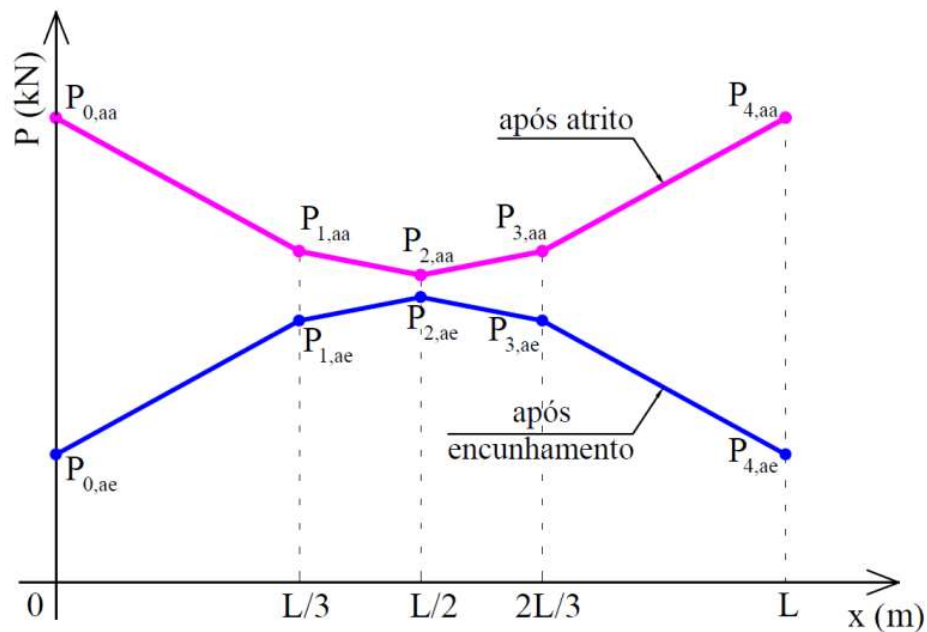
$$P_{01} = 2P_2 - P_0 - 2\Delta P \quad (2.23)$$

$$P_{11} = 2P_2 - P_1 - 2\Delta P \quad (2.24)$$

$$P_{21} = P_2 - 2\Delta P \quad (2.25)$$

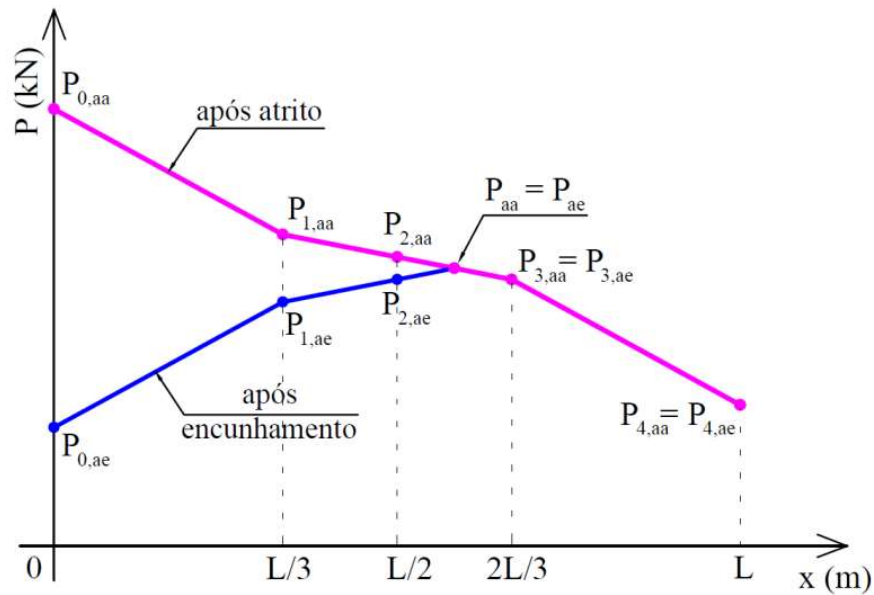
A seguir, é apresentado o desenvolvimento de perdas de protensão após atrito e encunhamento em um perfil de ancoragem ativa/ativa e ativa/passiva (ver Figura 2-10 e Figura 2-11).

Figura 2-10 – Perdas após atrito e encunhamento – ancoragens ativa/ativa



Fonte: Adaptada de Luchi (2016).

Figura 2-11 – Perdas após atrito e encunhamento – ancoragens ativa/passiva



Fonte: Adaptada de Luchi (2016).

2.1.3.3 Perdas de protensão por encurtamento imediato do concreto

De acordo com a NBR 6118:2014, a perda média na tensão de protensão por cabo pode ser calculada pela equação (2.26)

$$\Delta\sigma_p = \frac{\alpha_p(\sigma_{cp} + \sigma_{cg})(n - 1)}{2n} \quad (2.26)$$

em que

σ_{cp} é a tensão inicial no concreto ao nível do baricentro da armadura de protensão, devida à protensão simultânea de n cabos;

σ_{cg} é a tensão no concreto ao nível do baricentro da armadura de protensão, devida à carga permanente mobilizada pela protensão ou simultaneamente aplicada com a protensão;

α_p é relação entre o módulo de elasticidade do aço de armadura ativa E_p e o módulo de elasticidade inicial do concreto E_{ci} ;

n é o número de grupos de cabos protendidos simultaneamente.

2.1.3.4 Perdas progressivas por retração e fluência do concreto e relaxação do aço

O concreto é um material sujeito à retração e fluência, fenômenos ligados a perdas de protensão ao longo do tempo. A retração está relacionada com o equilíbrio higroscópico entre o concreto e o meio ambiente, levando o concreto a sofrer encurtamento à medida que perde

umidade. Já a fluência, está ligada a existência de ações de longa duração, que provocam deformações elásticas e plásticas progressivas nas fibras solicitadas. Como a protensão consiste na aplicação prévia de compressão na peça de concreto, e por se tratar de um carregamento permanente, o elemento protendido sofre encurtamento e conseqüente redução progressiva da força inicial de tração nos cabos, a se estabilizar num tempo infinito. Entretanto, na prática, observa-se que a maior parte dessas perdas ocorre nos dois a três primeiros anos.

Além desses dois fenômenos, soma-se a relaxação do aço, que consiste na redução de tensão na armadura, quando inicialmente alongada e mantida a comprimento constante. E, portanto, também resulta em perda lenta de protensão.

a) Deformação por fluência

A deformação por fluência do concreto ε_{cc} tem uma parcela rápida e irreversível ε_{ca} , que ocorre nas primeiras 24 horas, e outra lenta, ainda subdividida em reversível ε_{ccf} e irreversível ε_{ccd} . A NBR 6118:2014 esclarece que existem algumas hipóteses para o cálculo dos efeitos da fluência. São elas:

- A deformação por fluência varia linearmente com a tensão aplicada.
- Para acréscimos de tensão aplicados em instantes distintos, os respectivos efeitos de fluência se superpõem.
- A deformação rápida produz deformações constantes ao longo do tempo, os valores do coeficiente de deformação rápida φ_d são em função da relação entre a resistência do concreto no momento da aplicação da carga e sua resistência final.
- O coeficiente de deformação lenta reversível φ_d depende apenas da duração do carregamento e o seu valor final e o seu desenvolvimento ao longo do tempo são independentes da idade do concreto no momento da aplicação da carga.
- O coeficiente de deformação lenta irreversível φ_f depende da umidade relativa do ambiente (U), da consistência do concreto no lançamento, da espessura fictícia da peça, da idade fictícia do concreto no instante (t_0) da aplicação da carga e no instante considerado (t).

- Para o mesmo concreto, as curvas de deformação lenta irreversível em função do tempo, correspondentes às diferentes idades do concreto no carregamento, são obtidas umas em relação às outras, por deslocamento paralelo ao eixo das deformações.

A seguir serão apresentadas as equações da NBR 6118:2014 para idade fictícia e espessura fictícia da peça, necessárias ao cálculo da deformação devida à fluência. A idade fictícia t , expressa em dias, é dada por (2.27)

$$t = \alpha \sum_i \frac{T_i + 10}{30} \Delta t_{ef,i} \quad (2.27)$$

em que, α é o coeficiente dependente da velocidade de endurecimento do cimento, que na falta de dados experimentais, deve-se usar os valores do Quadro 2.1;

T_i é a temperatura média diária do ambiente em graus Celsius, e $\Delta t_{ef,i}$ é o período, em dias, em que T_i pode ser considerada constante.

Quadro 2.1 – Valores da fluência e da retração em função da velocidade de endurecimento do concreto

Cimento Portland (CP)	α	
	Fluência	Retração
De endurecimento lento (CP III e CP IV, todas as classes de resistência)	1	1
De endurecimento normal (CP I e CP II, todas as classes de resistência)	2	
De endurecimento rápido (CP V-ARI)	3	
Legenda: CP I e CP I-S – Cimento Portland comum CP II-E, CP II-F e CP II-Z – Cimento Portland composto CP III – Cimento Portland de alto forno CP IV – Cimento Portland pozolânico CP V-ARI – Cimento Portland de alta resistência inicial RS – Cimento Portland resistente a sulfatos (propriedade específica de alguns dos tipos de cimento citados)		

Fonte: NBR 6118:2014.

E a espessura fictícia da peça h_{fic} , expressa em centímetros, é dada por (2.28):

$$h_{fic} = \gamma \frac{2A_c}{u_{ar}} \quad (2.28)$$

em que, A_c é a área da seção transversal da peça, u_{ar} é a parte do perímetro externo da seção transversal da peça em contato com o ar, e γ é o coeficiente dependente da umidade relativa (U%) do ambiente definido pela equação (2.29)

$$\gamma = 1 + \exp(-7,8 + 0,1U) \quad (2.29)$$

Assim, no instante t , a deformação devida à fluência é dada pela equação (2.30)

$$\varepsilon_{cc}(t, t_0) = \varepsilon_{cca} + \varepsilon_{ccd} + \varepsilon_{ccf} = \frac{\sigma_c}{E_{ci28}} \varphi(t, t_0) \quad (2.30)$$

em que σ_c é a tensão no concreto, E_{ci28} é o módulo de deformação tangente inicial para $j=28$ dias, t é a idade fictícia do concreto no instante considerado, t_0 é a idade fictícia ao ser realizado o carregamento único, expressos em dias, e o coeficiente de fluência $\varphi(t, t_0)$ é dado pela expressão (2.31), cujos itens serão descritos adiante.

$$\varphi(t, t_0) = \varphi_a + \varphi_{f\infty} [\beta_f(t) - \beta_f(t_0)] + \varphi_{d\infty} \beta_d \quad (2.31)$$

O coeficiente de deformação rápida φ_a é obtido pelas expressões (2.32) e (2.33) para classes C20 a C45 e C50 a C90, respectivamente.

$$\varphi_a = 0,8 \left[1 - \frac{f_c(t_0)}{f_c(t_\infty)} \right] \quad (2.32)$$

$$\varphi_a = 1,4 \left[1 - \frac{f_c(t_0)}{f_c(t_\infty)} \right] \quad (2.33)$$

E $f_c(t_0)/f_c(t_\infty)$ deve ser calculado conforme (2.34)

$$\frac{f_c(t_0)}{f_c(t_\infty)} = \exp \left\{ s \left[1 - (28/t)^{1/2} \right] \right\} \quad (2.34)$$

sendo

$s = 0,38$ para concreto de cimento CPIII e IV;

$s = 0,25$ para concreto de cimento CPI e II;

$s = 0,20$ para concreto de cimento CPV-ARI;

e t é a idade efetiva do concreto em dias.

Em seguida, determina-se o valor final do coeficiente de deformação lenta irreversível pelas equações (2.35) e (2.36), para classes de concreto C20 a C45 e C50 a C90, respectivamente.

$$\varphi_{f\infty} = \varphi_{1c} \cdot \varphi_{2c} \quad (2.35)$$

$$\varphi_{f\infty} = 0,45 \cdot \varphi_{1c} \cdot \varphi_{2c} \quad (2.36)$$

Em que

φ_{1c} é o coeficiente dependente da umidade relativa do ambiente U (%) e da consistência do concreto (ver Quadro 2.2);

φ_{2c} é o coeficiente dependente da espessura fictícia da peça, obtido por (2.37)

$$\varphi_{2c} = \frac{42 + h_{fic}}{20 + h_{fic}} \quad (2.37)$$

$\beta_f(t)$ é o coeficiente relativo à deformação lenta irreversível, função da idade do concreto, definido por (2.38)

$$\beta_f(t) = \frac{t^2 + At + B}{t^2 + Ct + D} \quad (2.38)$$

sendo t ($t \geq 3$) o tempo expresso em dias, h a espessura fictícia em metros (para valores fora do intervalo $0,05 \leq h \leq 1,6$ adotar os extremos correspondentes), e A, B, C e D conforme as expressões (2.39) a (2.42)

$$A = 42h^3 - 350h^2 + 588h + 113 \quad (2.39)$$

$$B = 768h^3 - 3060h^2 + 3234h - 23 \quad (2.40)$$

$$C = -200h^3 + 13h^2 + 1090h + 183 \quad (2.41)$$

$$D = 7579h^3 - 31916h^2 + 35343h + 1931 \quad (2.42)$$

Quadro 2.2 – Valores numéricos usuais para a determinação da fluência e da retração

Ambiente	Umidade U %	Fluência $\varphi_{1c}^{a, c}$			Retração $10^4 \varepsilon_{1s}^{b, c}$			γ^d
		Abatimento de acordo com a ABNT NBR NM 67 cm						
		0 – 4	5 – 9	10 – 15	0 – 4	5 – 9	10 – 15	
Na água	–	0,6	0,8	1,0	+ 1,0	+ 1,0	+ 1,0	30,0
Em ambiente muito úmido imediatamente acima da água	90	1,0	1,3	1,6	– 1,9	– 2,5	– 3,1	5,0
Ao ar livre, em geral	70	1,5	2,0	2,5	– 3,8	– 5,0	– 6,2	1,5
Em ambiente seco	40	2,3	3,0	3,8	– 4,7	– 6,3	– 7,9	1,0

^a $\varphi_{1c} = 4,45 - 0,035U$ para abatimento no intervalo de 5 cm a 9 cm e $U \leq 90\%$.
^b $10^4 \varepsilon_{1s} = -8,09 + (U/15) - (U^2/2284) + (U^3/133765) - (U^4/7608150)$ para abatimentos de 5 cm a 9 cm e $40\% \leq U \leq 90\%$.
^c Os valores de φ_{1c} e ε_{1s} para $U \leq 90\%$ e abatimento entre 0 cm e 4 cm são 25 % menores e, para abatimentos entre 10 cm e 15 cm, são 25 % maiores.
^d $\gamma = 1 + \exp(-7,8 + 0,1 U)$ para $U \leq 90\%$.

NOTA 1 Para efeito de cálculo, as mesmas expressões e os mesmos valores numéricos podem ser empregados, no caso de tração.
 NOTA 2 Para o cálculo dos valores de fluência e retração, a consistência do concreto é aquela correspondente à obtida com o mesmo traço, sem a adição de superplastificantes e superfluidificantes.

Fonte: NBR 6118:2014.

Por fim, $\varphi_{d\infty}$ é o valor final do coeficiente de deformação reversível, considerado igual a 0,4. E $\beta_d(t)$ é o coeficiente relativo à deformação lenta reversível, função do tempo $(t - t_0)$ decorrido após o carregamento, obtido pela equação (2.43).

$$\beta_d(t) = \frac{t - t_0 + 20}{t - t_0 + 70} \quad (2.43)$$

b) Deformação por retração

A retração depende da umidade relativa do ambiente, da consistência do concreto em seu lançamento e da espessura fictícia da peça. Conforme a NBR 6118:2014, o valor da deformação por retração no intervalo de tempo t_0 a t é dado pela expressão (2.44)

$$\varepsilon_{cs}(t, t_0) = \varepsilon_{cs\infty} [\beta_s(t) - \beta_s(t_0)] \quad (2.44)$$

em que o valor final da retração $\varepsilon_{cs\infty}$ é descrito por (2.45)

$$\varepsilon_{cs\infty} = \varepsilon_{1s}\varepsilon_{2s} \quad (2.45)$$

sendo que o coeficiente dependente da umidade relativa do ambiente e da consistência do concreto ε_{1s} é obtido pelo Quadro 2.2 e o coeficiente dependente da espessura fictícia ε_{2s} é mostrado em (2.46)

$$\varepsilon_{2s} = \frac{33 + 2h_{fic}}{20,8 + 3h_{fic}} \quad (2.46)$$

E $\beta_s(t)$ ou $\beta_s(t_0)$ é o coeficiente relativo à retração, ver (2.47), no instante t ou t_0 , idades fictícias no instante considerado e no instante em que o efeito da retração começa a ser considerado, em dias.

$$\beta_s(t) = \frac{\left(\frac{t}{100}\right)^3 + A\left(\frac{t}{100}\right)^2 + B\left(\frac{t}{100}\right)}{\left(\frac{t}{100}\right)^3 + C\left(\frac{t}{100}\right)^2 + D\left(\frac{t}{100}\right) + E} \quad (2.47)$$

sendo t ($t \geq 3$) o tempo expresso em dias, h a espessura fictícia em metros (para valores fora do intervalo $0,05 \leq h \leq 1,6$ adotar os extremos correspondentes), e A, B, C, D e E conforme as expressões (2.48) a (2.52)

$$A = 40 \quad (2.48)$$

$$B = 116h^3 - 282h^2 + 220h - 4,8 \quad (2.49)$$

$$C = 2,5h^3 - 8,8h + 40,7 \quad (2.50)$$

$$D = -75h^3 + 585h^2 + 496h - 6,8 \quad (2.51)$$

$$E = -169h^4 + 88h^3 + 584h^2 - 39h + 0,8 \quad (2.52)$$

c) Relaxação do aço

Segundo a NBR 6118:2014, a perda de tensão por relaxação pura desde o instante t_0 do alongamento do cabo até o instante t considerado é obtida pela equação (2.53)

$$\Delta\sigma_{pr}(t, t_0) = \psi(t, t_0) \cdot \sigma_{pi} \quad (2.53)$$

em que σ_{pi} é a tensão na armadura ativa imediatamente após a aplicação da protensão e $\psi(t, t_0)$ é dado por (2.54)

$$\psi(t, t_0) = \psi_{1000} \left(\frac{t - t_0}{41,67}\right)^{0,15} \quad (2.54)$$

sendo ψ_{1000} o valor médio da relaxação, medido após 1000 horas, à temperatura constante de 20°C, para as perdas de tensão inicial de 50% a 80% de f_{ptk} , ver Quadro 2.3. Para tensões

menores do que $0,5 f_{ptk}$, considera-se que não há perda por relaxação, para tensões intermediárias às do Quadro 2.3, deve-se realizar interpolação e para o tempo infinito, ver (2.55).

$$\psi(t_{\infty}, t_0) = 2,5 \psi_{1000} \quad (2.55)$$

Quadro 2.3 – Valores de ψ_{1000} em porcentagem

σ_{po}	Cordoalhas		Fios		Barras
	RN	RB	RN	RB	
$0,5 f_{ptk}$	0	0	0	0	0
$0,6 f_{ptk}$	3,5	1,3	2,5	1,0	1,5
$0,7 f_{ptk}$	7,0	2,5	5,0	2,0	4,0
$0,8 f_{ptk}$	12,0	3,5	8,5	3,0	7,0
Onde RN é a relaxação normal; RB é a relaxação baixa.					

Fonte: NBR 6118:2014.

d) Cálculo das perdas progressivas por retração e fluência do concreto e relaxação do aço

No cálculo das perdas progressivas de protensão por retração e fluência do concreto e relaxação do aço, deve-se considerar a interação entre essas causas. A NBR 6118:2014 apresenta um processo simplificado para a determinação dessas perdas para o caso de fases únicas de operação. Admite que exista aderência entre a armadura e o concreto e que a peça permaneça no estágio I.

Ainda conforme a NBR 6118:2014, o processo simplificado é aplicado quando forem cumpridas as condições descritas abaixo:

- A concretagem e a protensão são executadas em fases próximas o suficiente para desprezar os efeitos recíprocos de fase uma sobre a outra.
- Os cabos apresentam espaçamentos suficientemente pequenos entre si, em relação à altura da peça, sendo possível considerar seus efeitos equivalentes ao de um único cabo, de seção igual à soma das áreas de todos os cabos, localizado na posição da resultante de seus esforços.

Assim, no tempo t , as perdas e deformações progressivas são obtidas pela equação (2.56). Consideram-se positivas as tensões de compressão no concreto e tração no aço.

$$\Delta\sigma_p(t, t_0) = \frac{\varepsilon_{cs}(t, t_0)E_p - \left(\frac{E_p}{E_{ci28}}\right)\sigma_{c,p0g}\varphi(t, t_0) - \sigma_{p0}\chi(t, t_0)}{1 + \chi(t, t_0) + [1 + 0,5\varphi(t, t_0)]\left(\frac{E_p}{E_{ci28}}\right)\left(1 + \frac{A_c}{I_c}e_p^2\right)\left(\frac{A_p}{A_c}\right)} \quad (2.56)$$

em que

$\varepsilon_{cs}(t, t_0)$ é o valor da deformação por retração no intervalo de tempo t_0 a t e é dado pela expressão (2.44);

E_p é o módulo de elasticidade da armadura ativa;

E_{ci28} é o módulo de deformação tangente inicial para $j=28$ dias;

$\sigma_{c,p0g}$ é a tensão no concreto adjacente ao cabo resultante, provocada pela protensão e pela carga permanente mobilizada no instante t_0 , sendo positiva para compressão;

$\varphi(t, t_0)$ é o coeficiente de fluência no concreto, dado pela expressão (2.31);

σ_{p0} é a tensão na armadura ativa devida à protensão e à carga permanente mobilizada no instante t_0 , sendo positiva para tração;

A_c é a área da seção transversal do concreto;

I_c é o momento central de inércia na seção do concreto;

e_p é a excentricidade do cabo resultante em relação ao baricentro da seção do concreto;

A_p é a área da seção transversal do cabo resultante;

e $\chi(t, t_0)$ é o coeficiente de fluência do aço conforme a equação (2.57)

$$\chi(t, t_0) = -\ln[1 - \psi(t, t_0)] \quad (2.57)$$

Em casos sem necessidade de grande precisão, os valores finais do coeficiente de fluência $\varphi(t_\infty, t_0)$ e deformação específica por retração $\varepsilon_{cs}(t_\infty, t_0)$ do concreto podem ser obtidos por interpolação. Esses valores são válidos para concretos plásticos e de cimento Portland comum, a temperaturas de 0°C a 40°C, sob tensões inferiores a 0,5 f_c (ao primeiro carregamento).

Quadro 2.4 – Valores característicos superiores da deformação específica de retração $\varepsilon_{cs}(t_{\infty}, t_0)$ e do coeficiente de fluência $\varphi(t_{\infty}, t_0)$

Umidade média ambiente %		40		55		75		90	
Espessura fictícia $2A_c/u$ cm		20	60	20	60	20	60	20	60
$\varphi(t_{\infty}, t_0)$ Concreto das classes C20 a C45	t_0 dias	5	4,6 3,8	3,9 3,3	2,8 2,4	2,0 1,9			
		30	3,4 3,0	2,9 2,6	2,2 2,0	1,6 1,5			
		60	2,9 2,7	2,5 2,3	1,9 1,8	1,4 1,4			
$\varphi(t_{\infty}, t_0)$ Concreto das classes C50 a C90	t_0 dias	5	2,7 2,4	2,4 2,1	1,9 1,8	1,6 1,5			
		30	2,0 1,8	1,7 1,6	1,4 1,3	1,1 1,1			
		60	1,7 1,6	1,5 1,4	1,2 1,2	1,0 1,0			
$\varepsilon_{cs}(t_{\infty}, t_0)$ ‰	t_0 dias	5	-0,53 -0,47	-0,48 -0,43	-0,36 -0,32	-0,18 -0,15			
		30	-0,44 -0,45	-0,41 -0,41	-0,33 -0,31	-0,17 -0,15			
		60	-0,39 -0,43	-0,36 -0,40	-0,30 -0,31	-0,17 -0,15			

Fonte: NBR 6118:2014.

2.1.4 Estados-limites de serviço e combinações

Dentre outros, a NBR 6118:2014 define os quatro estados-limites de serviço abaixo:

- a) Estado-limite de formação de fissuras ELS-F: inicia-se a formação de fissuras, que ocorre quando a tensão máxima de tração atinge a resistência do concreto à tração na flexão, dada pelas equações (2.58) e (2.59) para seções T e retangulares, respectivamente,

$$f_{ct,f} = 1,2 \cdot f_{ctk,inf} = 0,252 \cdot f_{ck}^{2/3} \quad (2.58)$$

$$f_{ct,f} = 1,5 \cdot f_{ctk,inf} = 0,315 \cdot f_{ck}^{2/3} \quad (2.59)$$

- b) Estado-limite de abertura das fissuras ELS-W: aquele em que as fissuras se apresentam com aberturas de valores iguais aos máximos especificados, sendo que as aberturas devem ser calculadas para a armadura passiva e ativa aderente, excluindo cabos em bainhas.
- c) Estado-limite de descompressão ELS-D: aquele em que um ou mais pontos da seção transversal atingem tensão nula, sem que haja tensão de tração no restante da seção.

- d) Estado-limite de compressão excessiva ELS-CE: no qual as tensões de compressão atingem o limite convencional estabelecido.

Hanai (2005) explica que, quando o concreto é submetido a tensões de compressão superiores a 50% de sua resistência, o efeito de microfissuração interna por compressão, antes discreto, é acentuado. Ocorre o aumento da microfissuras a partir de falhas na pasta de cimento e na interface com os agregados, responsável pelo comportamento não linear do concreto e efeito Rüsçh, quando a microfissuração fica instável com tensões acima de 70% da resistência à compressão. Assim, é interessante que no ELS-CE a tensão máxima de compressão no concreto seja limitada a $0,6 f_{ck}$.

Nos estados-limites de descompressão e formação de fissuras, o cálculo das tensões é realizado no estágio I: concreto não fissurado, comportamento elástico-linear dos materiais. Já no estado-limite de abertura das fissuras, o cálculo das tensões é realizado no estágio II.

Há três combinações de ações em serviço, apresentadas no Quadro 2.5. Elas são classificadas conforme sua permanência na estrutura. As combinações quase permanentes podem atuar durante grande parte da vida útil da estrutura, enquanto as frequentes se repetem muitas vezes ao longo da vida útil e as raras acontecem algumas vezes durante a vida da estrutura. Os valores dos coeficientes de redução ψ_1 e ψ_2 são obtidos no Quadro 2.6.

Quadro 2.5 – Combinações de serviço

Combinações de serviço (ELS)	Descrição	Cálculo das solicitações
Combinações quase permanentes de serviço (CQP)	Nas combinações quase permanentes de serviço, todas as ações variáveis são consideradas com seus valores quase permanentes $\psi_2 F_{qk}$	$F_{d,ser} = \Sigma F_{gi,k} + \Sigma \psi_2 F_{qj,k}$
Combinações frequentes de serviço (CF)	Nas combinações frequentes de serviço, a ação variável principal F_{q1} é tomada com seu valor frequente $\psi_1 F_{q1k}$ e todas as demais ações variáveis são tomadas com seus valores quase permanentes $\psi_2 F_{qk}$	$F_{d,ser} = \Sigma F_{gik} + \psi_1 F_{q1k} + \Sigma \psi_2 F_{qjk}$
Combinações raras de serviço (CR)	Nas combinações raras de serviço, a ação variável principal F_{q1} é tomada com seu valor característico F_{q1k} e todas as demais ações são tomadas com seus valores frequentes $\psi_1 F_{qk}$	$F_{d,ser} = \Sigma F_{gik} + F_{q1k} + \Sigma \psi_1 F_{qjk}$
<p>onde</p> <p>$F_{d,ser}$ é o valor de cálculo das ações para combinações de serviço;</p> <p>F_{q1k} é o valor característico das ações variáveis principais diretas;</p> <p>ψ_1 é o fator de redução de combinação frequente para ELS;</p> <p>ψ_2 é o fator de redução de combinação quase permanente para ELS.</p>		

Fonte: NBR 6118:2014.

Quadro 2.6 – Valores do coeficiente γ_{f2}

Ações		γ_{f2}		
		ψ_0	ψ_1^a	ψ_2
Cargas acidentais de edifícios	Locais em que não há predominância de pesos de equipamentos que permanecem fixos por longos períodos de tempo, nem de elevadas concentrações de pessoas ^b	0,5	0,4	0,3
	Locais em que há predominância de pesos de equipamentos que permanecem fixos por longos períodos de tempo, ou de elevada concentração de pessoas ^c	0,7	0,6	0,4
	Biblioteca, arquivos, oficinas e garagens	0,8	0,7	0,6
Vento	Pressão dinâmica do vento nas estruturas em geral	0,6	0,3	0
Temperatura	Variações uniformes de temperatura em relação à média anual local	0,6	0,5	0,3
<p>^a Para os valores de ψ_1 relativos às pontes e principalmente para os problemas de fadiga, ver Seção 23.</p> <p>^b Edifícios residenciais.</p> <p>^c Edifícios comerciais, de escritórios, estações e edifícios públicos.</p>				

Fonte: NBR 6118:2014.

2.1.5 Níveis de protensão

A NBR 6118:2014 determina três níveis de protensão: completa, limitada e parcial, relacionados aos estados-limites de serviço apresentados anteriormente e classes de agressividade ambiental.

a) Protensão completa:

É o nível máximo de protensão, estabelecido para a pré-tração nas classes de agressividade ambiental III e IV. Deve-se verificar o estado-limite de serviço de formação de fissuras para a combinação rara e o estado-limite de serviço de descompressão para a combinação frequente.

b) Protensão limitada:

Possui tensões de compressão provocadas pela protensão menores que as do nível anterior. É estipulada para a pré-tração na classe de agressividade ambiental II e pós-tração nas classes III e IV. Deve-se verificar o estado-limite de serviço de formação de fissuras para a combinação frequente e o estado-limite de serviço de descompressão para a combinação quase permanente.

c) Protensão parcial:

É o nível menos intenso de protensão. Especificado para a pré-tração na classe de agressividade ambiental I e pós-tração nas classes I e II. Deve-se verificar o estado-limite de serviço de abertura das fissuras, com valor limite de 0,2 mm, para a combinação frequente.

Nos três níveis de protensão, deve ser verificado o ELS-CE com tensão máxima de compressão no concreto limitada a $0,6 f_{ck}$.

A seguir, o Quadro 2.7 e o Quadro 2.8 apresentam, respectivamente, as classes de agressividade ambiental e exigências de durabilidade relacionadas à fissuração e à proteção da armadura.

Quadro 2.7- Classes de agressividade ambiental

Classe de agressividade ambiental	Agressividade	Classificação geral do tipo de ambiente para efeito de projeto	Risco de deterioração da estrutura
	I	Fraca	Rural Submersa
II	Moderada	Urbana ^{a, b}	Pequeno
III	Forte	Marinha ^a Industrial ^{a, b}	Grande
IV	Muito forte	Industrial ^{a, c} Respingos de maré	Elevado

^a Pode-se admitir um microclima com uma classe de agressividade mais branda (uma classe acima) para ambientes internos secos (salas, dormitórios, banheiros, cozinhas e áreas de serviço de apartamentos residenciais e conjuntos comerciais ou ambientes com concreto revestido com argamassa e pintura).

^b Pode-se admitir uma classe de agressividade mais branda (uma classe acima) em obras em regiões de clima seco, com umidade média relativa do ar menor ou igual a 65 %, partes da estrutura protegidas de chuva em ambientes predominantemente secos ou regiões onde raramente chove.

^c Ambientes quimicamente agressivos, tanques industriais, galvanoplastia, branqueamento em indústrias de celulose e papel, armazéns de fertilizantes, indústrias químicas.

Fonte: NBR 6118:2014.

Quadro 2.8 - Exigências de durabilidade relacionadas à fissuração e à proteção da armadura, em função das classes de agressividade ambiental

Tipo de concreto estrutural	Classe de agressividade ambiental (CAA) e tipo de protensão	Exigências relativas à fissuração	Combinação de ações em serviço a utilizar
Concreto simples	CAA I a CAA IV	Não há	–
Concreto armado	CAA I	ELS-W $w_k \leq 0,4$ mm	Combinação frequente
	CAA II e CAA III	ELS-W $w_k \leq 0,3$ mm	
	CAA IV	ELS-W $w_k \leq 0,2$ mm	
Concreto protendido nível 1 (protensão parcial)	Pré-tração com CAA I ou Pós-tração com CAA I e II	ELS-W $w_k \leq 0,2$ mm	Combinação frequente
Concreto protendido nível 2 (protensão limitada)	Pré-tração com CAA II ou Pós-tração com CAA III e IV	Verificar as duas condições abaixo	
		ELS-F	Combinação frequente
		ELS-D ^a	Combinação quase permanente
Concreto protendido nível 3 (protensão completa)	Pré-tração com CAA III e IV	Verificar as duas condições abaixo	
		ELS-F	Combinação rara
		ELS-D ^a	Combinação frequente

^a A critério do projetista, o ELS-D pode ser substituído pelo ELS-DP com $a_p = 50$ mm (Figura 3.1).

NOTAS

1 As definições de ELS-W, ELS-F e ELS-D encontram-se em 3.2.

2 Para as classes de agressividade ambiental CAA-III e IV, exige-se que as cordoalhas não aderentes tenham proteção especial na região de suas ancoragens.

3 No projeto de lajes lisas e cogumelo protendidas, basta ser atendido o ELS-F para a combinação frequente das ações, em todas as classes de agressividade ambiental.

Fonte: NBR 6118:2014.

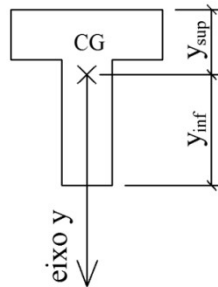
2.1.6 Determinação da protensão

Sejam consideradas negativa a tensão de compressão e positiva a de tração. A Figura 2-12 apresenta a convenção adotada para o eixo y , e definição de y_{sup} e y_{inf} como distâncias do centro de gravidade CG da seção transversal até a borda superior e inferior, respectivamente. Observa-se que y_{sup} é negativo e y_{inf} é positivo. Portanto, o módulo resistente à flexão superior W_{sup} é negativo e o inferior W_{inf} é positivo, como mostram as equações (2.60) e (2.61), em que, I_x é o momento de inércia em relação ao eixo x .

$$W_{sup} = \frac{I_x}{y_{sup}} \quad (2.60)$$

$$W_{inf} = \frac{I_x}{y_{inf}} \quad (2.61)$$

Figura 2-12 – Convenção adotada para o eixo y na seção transversal



Fonte: Elaborada pela autora.

Assim, as tensões nas bordas superior e inferior são obtidas nas equações (2.62) e (2.63)

$$\sigma_{inf} = \frac{M_g}{W_{inf}} + \frac{\psi \cdot M_q}{W_{inf}} + \frac{P}{A} + \frac{P \cdot e}{W_{inf}} \quad (2.62)$$

$$\sigma_{sup} = \frac{M_g}{W_{sup}} + \frac{\psi \cdot M_q}{W_{sup}} + \frac{P}{A} + \frac{P \cdot e}{W_{sup}} \quad (2.63)$$

em que

M_g é o momento devido às cargas permanentes;

M_q é o momento devido às cargas acidentais;

ψ é o coeficiente de redução de combinação frequente ou quase-permanente;

P é a força de protensão;

A é a área da seção transversal;

e é a excentricidade dos cabos, ou seja, a distância entre o centro de gravidade dos cabos e o CG da seção transversal da peça.

As tensões na seção da peça devem atender aos estados-limites de serviço em suas devidas combinações, conforme o nível de protensão exigido. Assim, é realizado o pré-dimensionamento da força necessária nos cabos. Em seguida, arbitra-se um valor para as perdas de protensão, da ordem de 20% a 30% do valor final estimado.

O processo de determinação da força de protensão acima exposto aplica-se bem para vigas isostáticas nos níveis de protensão completa e limitada. Para os casos de protensão parcial, torna-se interessante o método do balanceamento das cargas, que consiste no cálculo da força de protensão necessária e traçado dos cabos em elevação para equilibrar certa parcela do carregamento externo atuante.

Encontrado o valor inicial da protensão e definidos o número e o traçado dos cabos, podem ser calculadas as perdas imediatas e progressivas, obtendo-se a força real na armadura ativa. Pode-se, assim, verificar os estados-limites de serviço nas seções ao longo da peça e, finalmente, verificar o estado-limite último.

2.1.7 Estado-limite último no ato da protensão

A aplicação da protensão deve ser permitida somente após comprovação experimental de que a menor resistência efetiva à compressão da peça já tenha atingido a resistência característica do concreto correspondente à idade fictícia j (f_{ckj}), especificado em projeto.

Na verificação do estado-limite último no ato da protensão, devem ser considerados apenas os carregamentos que efetivamente atuarem na ocasião da aplicação de tensão na armadura ativa, usualmente o peso próprio. A NBR 6118:2014 especifica os seguintes valores dos coeficientes de ponderação a serem utilizados para:

- resistência do concreto $\gamma_c = 1,2$;
- resistência do aço $\gamma_s = 1,15$;
- cargas oriundas da protensão na pré-tração $\gamma_p = 1,0$;
- cargas oriundas da protensão na pós-tração $\gamma_p = 1,1$;

- ações desfavoráveis $\gamma_f = 1,0$;
- ações favoráveis $\gamma_f = 0,9$.

A NBR 6118:2014 permite uma verificação simplificada, em que se admite a segurança em relação ao estado-limite último no ato da protensão verificada no estágio I, desde que sejam satisfeitas as condições a seguir:

- a) A tensão máxima de compressão na seção de concreto, obtida com as solicitações ponderadas de $\gamma_p = 1,1$ e $\gamma_f = 1,0$, não pode ultrapassar 70% da resistência característica prevista para a idade de aplicação da protensão, conforme a equação (2.64)

$$\sigma_{c,máx} \leq 0,7f_{ckj} \quad (2.64)$$

- b) A tensão máxima de tração não pode ultrapassar 120% da resistência média à tração no concreto correspondente ao f_{ckj} especificado, ver expressão (2.65)

$$\sigma_{t,máx} \leq 1,2f_{ctm} = 0,36f_{ckj}^{2/3} \quad (2.65)$$

- c) Quando houver tensões de tração na seção transversal, deve haver armadura de tração calculada no estágio II. Para efeito de cálculo na fase da construção, a força nessa armadura pode ser considerada igual à resultante de tração no estágio I e o acréscimo de tensão devido a essa força fica limitada a 150 MPa para fios e barras lisas e 250 MPa para barras nervuradas.

2.1.8 Estado-limite último

O estado-limite último está relacionado ao colapso ou a qualquer outra forma de ruína estrutural que determine a sua paralisação, no todo ou em parte. A NBR 6118:2014 admite algumas hipóteses na verificação do estado-limite último de vigas:

- As seções transversais permanecem planas após a deformação.
- A deformação das armações passivas ou o acréscimo de deformação das ativas aderentes em tração ou compressão deve ser igual a do concreto do entorno.
- Para as armaduras ativas não aderentes, caso não haja valores experimentais ou análises não lineares adequadas, os valores do acréscimo de tensões para estruturas usuais de edifícios seguem abaixo, devendo ser divididos por seus coeficientes de ponderação.
 - Para elementos com relação vão/altura útil ≤ 35 , ver (2.66)

$$\Delta\sigma_p = 70 + \frac{f_{ck}}{100\rho_p} \leq 420MPa \quad (2.66)$$

- Para elementos com relação vão/altura útil > 35 , ver (2.67)

$$\Delta\sigma_p = 70 + \frac{f_{ck}}{300\rho_p} \leq 210MPa \quad (2.67)$$

em que

$\Delta\sigma_p$ e f_{ck} são dados em MPa;

ρ_p é a taxa geométrica da armadura ativa, conforme a expressão (2.68);

$$\rho_p = \frac{A_p}{b_c d_p} \quad (2.68)$$

sendo

b_c a largura da mesa de compressão;

d_p a altura útil referida à armadura ativa.

- d) As tensões de tração no concreto, perpendiculares à seção transversal, devem ser desprezadas no ELU.
- e) A distribuição de tensões no concreto pode ser simplificada pelo retângulo de profundidade y dada por (2.69)

$$y = \lambda \cdot x \quad (2.69)$$

sendo λ dado por (2.70) para $f_{ck} \leq 50$ MPa;

$$\lambda = 0,8 \quad (2.70)$$

e a tensão constante no concreto atuante até a profundidade y é dada por (2.71), caso a largura da seção, medida paralelamente à linha neutra, não diminuir a partir desta para a borda comprimida, e dada por (2.72) caso contrário;

$$\sigma_c = \alpha_c f_{cd} \quad (2.71)$$

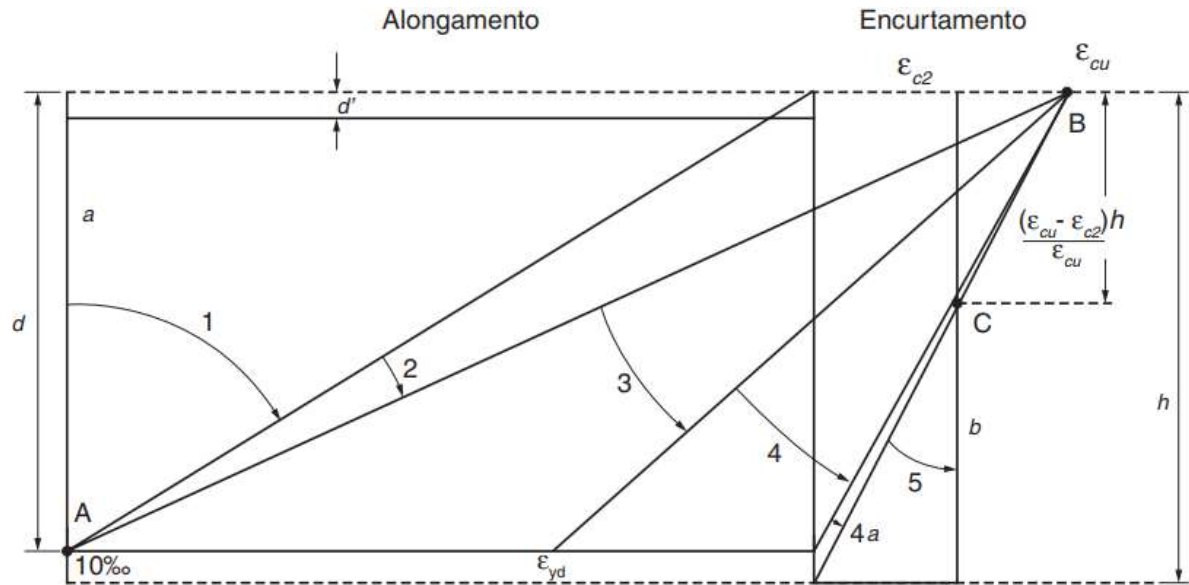
$$\sigma_c = 0,9\alpha_c f_{cd} \quad (2.72)$$

em que α_c é definido pela equação (2.73) para $f_{ck} \leq 50$ MPa;

$$\alpha_c = 0,85 \quad (2.73)$$

- f) A tensão nas armaduras deve ser obtida a partir dos diagramas tensão-deformação.
- g) O estado-limite último é caracterizado quando a distribuição das deformações na seção transversal pertencer a um dos domínios da Figura 2-13.

Figura 2-13 – Domínios de estado-limite último de uma seção transversal



Fonte: NBR 6118:2014.

Em que a deformação específica de encurtamento do concreto no início do patamar plástico ε_{c2} e a deformação específica do concreto por encurtamento na ruptura ε_{cu} são dadas por (2.74) e (2.75) para $f_{ck} \leq 50$ MPa;

$$\varepsilon_{c2} = 2,0\text{‰} \quad (2.74)$$

$$\varepsilon_{cu} = 3,5\text{‰} \quad (2.75)$$

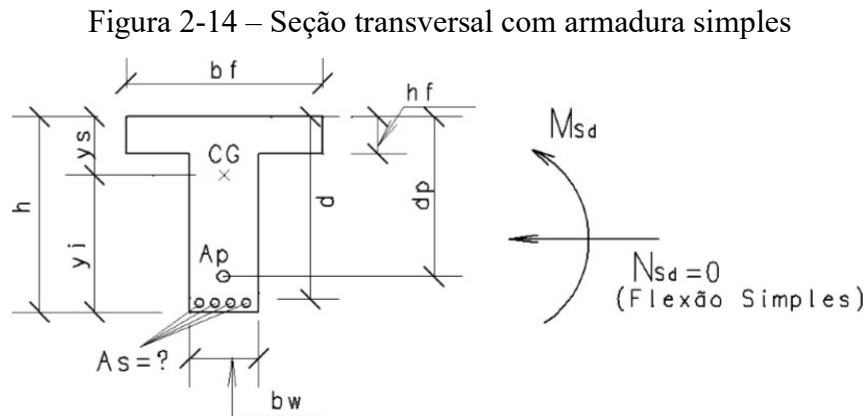
Outro aspecto importante é a necessidade de garantir boas condições de ductilidade em vigas e lajes, para isso, deve-se respeitar o limite da relação entre posição da linha neutra (x) em relação à altura útil da peça (d) de $x/d \leq 0,45$ para concretos de $f_{ck} \leq 50$ MPa. Se necessário, deve-se adotar armadura de compressão para garantir a ductilidade da peça.

Os esforços nas armaduras, ativas e passivas, podem ser considerados em seus centros de gravidade, desde que a armadura mais afastada deste não ultrapasse 10% da altura da peça. As armaduras laterais podem ser consideradas na resistência aos esforços, desde que estejam adequadamente ancoradas e emendadas.

Ainda segundo a NBR 6118:2014, no estado-limite último, podem ser considerados apenas os esforços solicitantes hiperestáticos de protensão, além de outras ações. Os isostáticos de protensão não podem ser incluídos. Os cabos são levados em conta a partir dos diagramas

tensão-deformação e análise de seus pré-alongamentos. Esses devem ser calculados usando as tensões iniciais com valores de cálculo e perdas na idade analisada.

A Figura 2-14 mostra a seção transversal de uma viga T protendida submetida à flexão simples,



Fonte: Luchi (2016).

em que

h é a altura total da peça;

h_f é a altura da mesa.

b_f é a largura da mesa;

b_w é a largura da alma;

y_s é a distância da face superior ao centro de gravidade da seção transversal;

y_i é a distância da face inferior ao centro de gravidade da seção transversal;

d_p é a altura útil da armadura ativa;

d é a altura útil da armadura passiva;

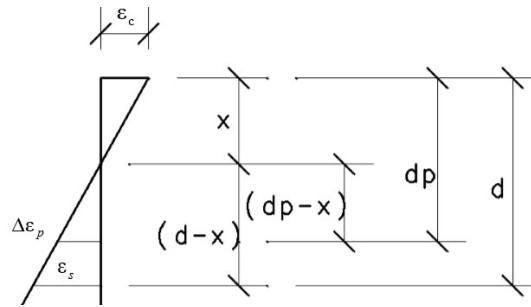
M_{sd} é o momento fletor solicitante;

A_p é a área da seção transversal armadura ativa;

A_s é a área da seção transversal armadura passiva.

O dimensionamento da peça à flexão é realizado a partir do equilíbrio dos esforços na seção. São conhecidos o momento fletor solicitante, a geometria da peça, e os dados referentes à armadura ativa. Assim, deve-se arbitrar o valor para x/d e calcular as deformações correspondentes aos domínios de estado-limite último, ver Figura 2-15,

Figura 2-15 – Deformada no ELU



Fonte: Luchi (2016).

em que

x é a posição da linha neutra;

ε_c é a deformação na fibra mais comprimida do concreto;

$\Delta\varepsilon_p$ é o alongamento adicional ao pré-alongamento da armadura ativa;

ε_s é o alongamento da armadura passiva de tração.

Para concretos até a classe C50, se $0 \leq x/d \leq 0,259$, a deformação no concreto é dada pela equação (2.76)

$$\varepsilon_c = 10\text{‰} \cdot \frac{x}{d-x} \quad (2.76)$$

Se $0,259 \leq x/d \leq h/d$, a deformação no concreto é dada pela expressão (2.77)

$$\varepsilon_c = 3,5\text{‰} \quad (2.77)$$

E se $x/d \geq h/d$, a deformação no concreto é dada por (2.78)

$$\varepsilon_c = \frac{2\text{‰}}{1 - \frac{3}{7} \cdot \frac{h}{x}} \quad (2.78)$$

Conhecidos x e ε_c , pode-se encontrar $\Delta\varepsilon_p$ e ε_s por semelhança de triângulos na Figura 2-14, conforme a equação (2.79)

$$\frac{\varepsilon_c}{x} = \frac{\Delta\varepsilon_p}{d_p - x} = \frac{\varepsilon_s}{d - x} \quad (2.79)$$

Na protensão não aderente, como ocorre o deslizamento das cordoalhas, o acréscimo de alongamento $\Delta\varepsilon_p$ não é o ganho na seção estudada, mas uma média desses ganhos ao longo de toda a viga.

O alongamento total na armadura ativa ε_p é obtido por (2.80)

$$\varepsilon_p = \varepsilon_{pré} + \Delta\varepsilon_p \quad (2.80)$$

em que $\varepsilon_{pré}$ é o pré-alongamento na armadura, dado por (2.81) para pré-tração e por (2.82) para pós-tração

$$\varepsilon_{pré} \cong \gamma_p \frac{P_\infty}{E_p A_p} \quad (2.81)$$

$$\varepsilon_{pré} \cong \gamma_p \left[\frac{P_\infty}{E_p A_p} (1 + \alpha_p \rho_p \eta) \right] \quad (2.82)$$

em que γ_p , α_p , ρ_p e η são dados por (2.83) a (2.86)

$$\gamma_p = 0,9 \quad (2.83)$$

$$\alpha_p = \frac{E_p}{E_{ci}} \quad (2.84)$$

$$\rho_p = \frac{A_p}{b_c d_p} \quad (2.85)$$

$$\eta = \left(1 + e_p^2 \cdot \frac{A_c}{I_c} \right) \quad (2.86)$$

Calculadas ε_s e ε_p , as tensões nas armaduras passivas e ativas podem ser obtidas pelas equações constitutivas (2.87) e (2.88)

$$-f_{yd} \leq \sigma_{sd} = E_s \cdot \varepsilon_s \leq f_{yd} \quad (2.87)$$

$$\sigma_{pd} = E_p \cdot \varepsilon_p \leq f_{pyd} \quad (2.88)$$

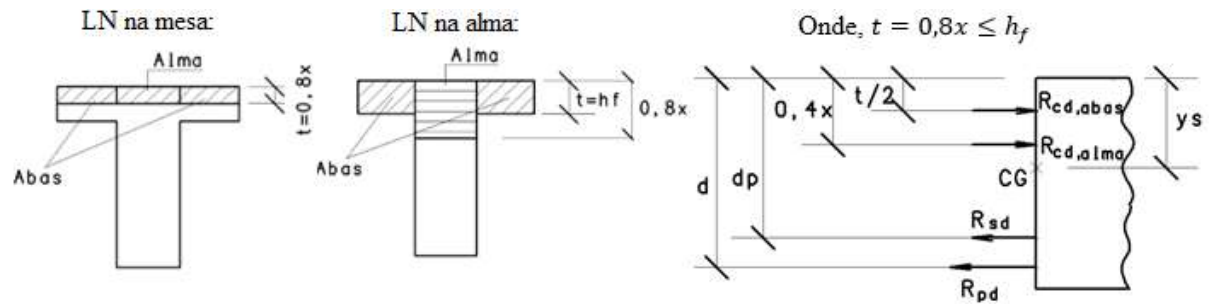
No caso de protensão não aderente, σ_{pd} é dado pela expressão (2.89)

$$\sigma_{pd} = E_p \cdot \varepsilon_{pré} + \Delta\sigma_p \quad (2.89)$$

em que $\Delta\sigma_p$ é calculado conforme as equações (2.66) e (2.67).

Assim, podem ser encontradas as resultantes nas armaduras e no concreto e seus pontos de aplicação na seção transversal, conforme a Figura 2-16.

Figura 2-16 - Resultantes na seção transversal



Fonte: Luchi (2016).

em que

R_{pd} é a força resultante de cálculo na armadura ativa;

R_{sd} é a força resultante de cálculo na armadura passiva.

$R_{cd,alma}$ é a força resultante de cálculo na alma;

$R_{cd,abas}$ é a força resultante de cálculo nas abas.

Essas resultantes são dadas pelas equações (2.90) a (2.93)

$$R_{pd} = \sigma_{pd} \cdot A_p \quad (2.90)$$

$$R_{sd} = \sigma_{sd} \cdot A_s \quad (2.91)$$

$$R_{cd,alma} = 0,85 f_{cd} \cdot b_w \cdot 0,8x \quad (2.92)$$

$$R_{cd,abas} = 0,85 f_{cd} \cdot (b_f - b_w) \cdot t \quad (2.93)$$

Portanto, os esforços resistentes são apresentados em (2.94) e (2.95)

$$N_{Rd} = R_{cd,alma} + R_{cd,abas} - R_{pd} - \sigma_{sd} \cdot A_s \quad (2.94)$$

$$M_{Rd} = R_{cd,alma}(y_s - 0,4x) + R_{cd,abas} \left(y_s - \frac{t}{2} \right) + \sigma_{sd} A_s (d - y_s) + R_{pd}(d_p - y_s) \quad (2.95)$$

Pela imposição do equilíbrio na seção, têm-se as equações (2.96) e (2.97)

$$N_{Rd} = R_{cd,alma} + R_{cd,abas} - R_{pd} - \sigma_{sd} \cdot A_s = 0 \quad (2.96)$$

$$M_{Rd} \geq M_{Sd} \quad (2.97)$$

Isolando A_s em (2.96), tem-se

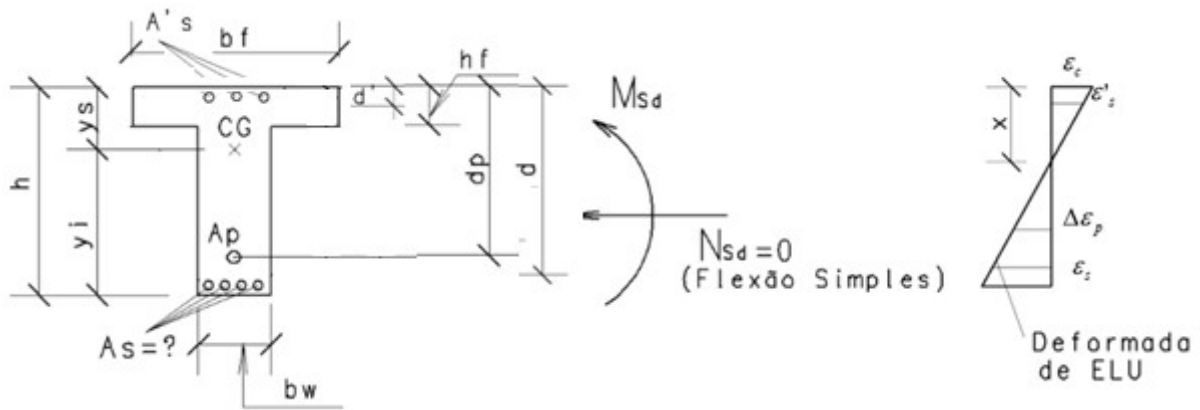
$$A_s = \frac{R_{cd,alma} + R_{cd,abas} - R_{pd}}{\sigma_{sd}} \quad (2.98)$$

Substituindo (2.98) em (2.95), obtém-se (2.99). O valor encontrado para M_{Rd} deve ser superior a M_{Sd} , caso não seja, deve-se repetir o processo até que o momento fletor resistente se iguale ao solicitante.

$$M_{Rd} = R_{cd,alma}(y_s - 0,4x) + R_{cd,abas}\left(y_s - \frac{t}{2}\right) + (R_{cd,alma} + R_{cd,abas} - R_{pd})(d - y_s) + R_{pd}(d_p - y_s) \quad (2.99)$$

Caso não tenha sido obtido sucesso e seja necessária a adoção de armadura passiva dupla (ver Figura 2-17), deve-se utilizar o valor máximo para x/d permitido pela NBR 6118:2014, repetir os passos anteriores e calcular também os alongamentos e tensões da armadura de compressão, dados por (2.100) e (2.101).

Figura 2-17 – Seção transversal com armadura passiva dupla e deformada



Fonte: Luchi (2016).

$$\varepsilon'_s = \varepsilon_c \frac{x - d'}{x} \quad (2.100)$$

$$-f_{yd} \leq \sigma'_{sd} = E_s \cdot \varepsilon'_s \leq f_{yd} \quad (2.101)$$

Então, têm-se os esforços resistentes e, impondo o equilíbrio, são obtidas as expressões (2.102) e (2.103)

$$N_{Rd} = R_{cd,alma} + R_{cd,abas} - R_{pd} - \sigma_{sd} \cdot A_s + \sigma'_{sd} \cdot A'_s = 0 \quad (2.102)$$

$$M_{Rd} = R_{cd,alma}(y_s - 0,4x) + R_{cd,abas}\left(y_s - \frac{t}{2}\right) + R_{pd}(d_p - y_s) + \sigma_{sd}A_s(d - y_s) + \sigma'_{sd}A'_s(y_s - d') \geq M_{Sd} \quad (2.103)$$

Assim, há duas equações e duas incógnitas, sendo possível determinar o par de A_s e A'_s para a razão x/d adotada. Pode-se utilizar o par de A_s e A'_s mais conveniente, desde que sejam atendidos os limites de x/d estabelecidos por norma.

Em todos os casos, devem-se respeitar as taxas mínimas de armaduras definidas pela NBR 6118:2014, conforme o Quadro 2.9 e o Quadro 2.10.

Quadro 2.9 – Valores mínimos para armaduras passivas aderentes

Armadura	Elementos estruturais sem armaduras ativas	Elementos estruturais com armadura ativa aderente	Elementos estruturais com armadura ativa não aderente
Armaduras negativas	$\rho_s \geq \rho_{\min}$	$\rho_s \geq \rho_{\min} - \rho_p \geq 0,67 \rho_{\min}$	$\rho_s \geq \rho_{\min} - 0,5 \rho_p \geq 0,67 \rho_{\min}$ (ver 19.3.3.2)
Armaduras negativas de bordas sem continuidade	$\rho_s \geq 0,67 \rho_{\min}$		
Armaduras positivas de lajes armadas nas duas direções	$\rho_s \geq 0,67 \rho_{\min}$	$\rho_s \geq 0,67 \rho_{\min} - \rho_p \geq 0,5 \rho_{\min}$	$\rho_s \geq \rho_{\min} - 0,5 \rho_p \geq 0,5 \rho_{\min}$
Armadura positiva (principal) de lajes armadas em uma direção	$\rho_s \geq \rho_{\min}$	$\rho_s \geq \rho_{\min} - \rho_p \geq 0,5 \rho_{\min}$	$\rho_s \geq \rho_{\min} - 0,5 \rho_p \geq 0,5 \rho_{\min}$
Armadura positiva (secundária) de lajes armadas em uma direção	$A_s/s \geq 20\%$ da armadura principal $A_s/s \geq 0,9 \text{ cm}^2/\text{m}$ $\rho_s \geq 0,5 \rho_{\min}$		–
onde $\rho_s = A_s/b_w h$ e $\rho_p = A_p/b_w h$.			
NOTA Os valores de ρ_{\min} são definidos em 17.3.5.2.1.			

Fonte: NBR 6118:2014.

Quadro 2.10 – Taxas mínimas de armadura de flexão

Forma da seção	Valores de $\rho_{\text{mín}}^a$ ($A_{s,\text{mín}}/A_c$) %														
	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90
Retangular	0,150	0,150	0,150	0,164	0,179	0,194	0,208	0,211	0,219	0,226	0,233	0,239	0,245	0,251	0,256

^a Os valores de $\rho_{\text{mín}}$ estabelecidos nesta Tabela pressupõem o uso de aço CA-50, $d/h = 0,8$ e $\gamma_c = 1,4$ e $\gamma_s = 1,15$. Caso esses fatores sejam diferentes, $\rho_{\text{mín}}$ deve ser recalculado.

Fonte: NBR 6118:2014.

2.1.9 Protensão parcial

De acordo com a NBR 6118:2014, o nível de protensão parcial produz tensões de compressão mais brandas do que a limitada e completa. Ocorre fissuração da peça, devendo ser respeitado o estado-limite de serviço de abertura das fissuras na combinação frequente. Assim, o valor máximo característico da abertura de fissuras é apresentado em (2.104).

$$w_{k,máx,CF} \leq 0,2 \text{ mm} \quad (2.104)$$

O critério para verificação da fissuração é o mesmo adotado para estruturas de concreto armado, as aberturas devem ser calculadas para armaduras passivas e ativas aderentes, excluindo os cabos em bainhas.

Após o dimensionamento no estado-limite último, são conhecidas as armaduras ativas e passivas, definidos o número de barras e cabos e o seu alojamento. Caso a seção tenha ultrapassado o estado-limite de serviço de formação de fissuras na combinação frequente, chegando ao estágio II, conforme a equação (2.105), está caracterizada a protensão parcial,

$$\sigma_{c,máx,CF} > f_{ct,f} = \alpha \times f_{ctk,inf} \quad (2.105)$$

em que $\alpha = 1,5$ para seções retangulares e $\alpha = 1,2$ para seções T, e $f_{ctk,inf}$ é dado por (2.106).

$$f_{ctk,inf} = 0,7 \times 0,3 \times f_{ck}^{2/3} \quad (2.106)$$

Deve-se encontrar o valor da tensão na armadura σ_s e posição da linha neutra x para a combinação frequente. No estágio II, considera-se o diagrama linear do concreto comprimido, desprezando-se a sua resistência à tração (ver Figura 2-18).

A NBR 6118:2014 explica que, no estágio II, a relação entre os módulos de elasticidade do aço e do concreto, chamada α_e pode ser considerado igual a quinze, ver (2.117)

$$\alpha_e = \frac{E_s}{E_c} = 15 \quad (2.117)$$

Portanto, pode-se escrever σ_c conforme (2.118)

$$\sigma_c = \frac{\sigma_s}{\alpha_e} \frac{x}{d-x} \quad (2.118)$$

Em seguida, podem ser relacionadas as equações das forças resultantes na seção. Ver equações (2.119) a (2.122).

$$N_c = \frac{1}{2} b_c x \sigma_c = \sigma_s \frac{b_c x}{2\alpha_e} \frac{x}{d-x} \quad (2.119)$$

$$N_s' = A_s' \sigma_s' = \sigma_s A_s' \frac{x-d'}{d-x} \quad (2.120)$$

$$N_p = \varepsilon_{pré} E_p A_p + \sigma_s A_p \frac{E_p}{E_s} \frac{d_p - x}{d-x} \quad (2.121)$$

$$N_s = A_s \sigma_s \quad (2.122)$$

Do equilíbrio das forças para flexão simples, ver (2.123) e (2.124), obtém-se σ_s em função de x , apresentado em (2.125). A equação (2.125) é válida para protensão aderente em seções retangulares ou seções T com linha neutra na mesa, em que b_c é igual a b_w ou b_f , respectivamente.

$$N_c + N_s' = N_p + N_s \quad (2.123)$$

$$\sigma_s \frac{b_c x}{2\alpha_e} \frac{x}{d-x} + \sigma_s A_s' \frac{x-d'}{d-x} = \varepsilon_{pré} E_p A_p + \sigma_s A_p \frac{E_p}{E_s} \frac{d_p - x}{d-x} + A_s \cdot \sigma_s \quad (2.124)$$

$$\sigma_s = \frac{\varepsilon_{pré} E_p A_p}{\frac{b_c \cdot x}{2\alpha_e} \frac{x}{d-x} + A_s' \frac{x-d'}{d-x} - A_s - \frac{E_p}{E_s} \frac{d_p - x}{d-x} A_p} \quad (2.125)$$

Para protensão não aderente, não há compatibilidade de deformações entre a armadura ativa e a seção de concreto, portanto, a parcela relativa à $\Delta\varepsilon_p$ é considerada nula, conforme (2.126).

$$\sigma_s = \frac{\varepsilon_{pré} E_p A_p}{\frac{b_c \cdot x}{2\alpha_e} \frac{x}{d-x} + A_s' \frac{x-d'}{d-x} - A_s} \quad (2.126)$$

Do equilíbrio dos momentos, chega-se as expressões (2.127) e (2.128), onde M_{CF} é o momento na combinação frequente, ver Figura 2-18.

A força N_c e momento M_c referentes ao concreto comprimido são dados pelas equações (2.131) e (2.132). Simplificando (2.132), chega-se a (2.133).

$$N_c = \sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} x b_f - \sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} \frac{(x-h_f)^2}{x} (b_f - b_w) \quad (2.131)$$

$$M_c = \sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} x b_f \left(y_{cg} - \frac{x}{3}\right) - \sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} \frac{(x-h_f)^2}{x} (b_f - b_w) \left(y_{cg} - \frac{x}{3} + \frac{h_f}{3}\right) \quad (2.132)$$

$$M_c = \sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} \left(x b_f \left(y_{cg} - \frac{x}{3}\right) - \frac{(x-h_f)^2}{x} (b_f - b_w) \left(y_{cg} - \frac{x}{3} + \frac{h_f}{3}\right) \right) \quad (2.133)$$

Do equilíbrio das forças, para flexão simples em seções T com linha neutra na alma, ver (2.134) e (2.135), obtém-se σ_s em função de x , apresentado em (2.136). A equação (2.136) é válida protensão aderente.

$$N_c + N_s' = N_p + N_s \quad (2.134)$$

$$\sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} \left(x b_f - \frac{(x-h_f)^2}{x} (b_f - b_w) \right) + \sigma_s A_s' \frac{x-d'}{d-x} = \quad (2.135)$$

$$= \varepsilon_{pré} E_p A_p + \sigma_s A_p \frac{E_p}{E_s} \frac{d_p - x}{d-x} + A_s \cdot \sigma_s$$

$$\sigma_s = \frac{\varepsilon_{pré} E_p A_p}{\frac{1}{2\alpha_e} \frac{x}{d-x} \left[x b_f - \frac{(x-h_f)^2}{x} (b_f - b_w) \right] + A_s' \frac{x-d'}{d-x} - A_s - \frac{E_p}{E_s} \frac{d_p - x}{d-x} A_p} \quad (2.136)$$

Para protensão não aderente, não há compatibilidade de deformações entre a armadura ativa e a seção de concreto, portanto, a parcela relativa à $\Delta\varepsilon_p$ é considerada nula, conforme (2.160).

$$\sigma_s = \frac{\varepsilon_{pré} E_p A_p}{\frac{1}{2\alpha_e} \frac{x}{d-x} \left[x b_f - \frac{(x-h_f)^2}{x} (b_f - b_w) \right] + A_s' \frac{x-d'}{d-x} - A_s} \quad (2.137)$$

Do equilíbrio dos momentos, para seções T com linha neutra na alma, chega-se as expressões (2.138) e (2.139), ver Figura 2-19.

$$M_c + N_s'(y_{cg} - d') + N_s(d - y_{cg}) + N_p(d_p - y_{cg}) = M_{CF} \quad (2.138)$$

$$\begin{aligned} \sigma_s \frac{1}{2\alpha_e} \frac{x}{d-x} \left(x b_f \left(y_{cg} - \frac{x}{3}\right) - \frac{(x-h_f)^2}{x} (b_f - b_w) \left(y_{cg} - \frac{x}{3} + \frac{h_f}{3}\right) \right) + \sigma_s A_s' \frac{x-d'}{d-x} (y_{cg} - d') + \\ + A_s \sigma_s (d - y_{cg}) + \left(\varepsilon_{pré} E_p A_p + \sigma_s A_p \frac{E_p}{E_s} \frac{d_p - x}{d-x} \right) (d_p - y_{cg}) = M_{CF} \end{aligned} \quad (2.139)$$

Ao isolar σ_s na equação (2.139), obtém-se a expressão (2.140), válida para protensão aderente.

$$\sigma_s = \frac{M_{CF} - \varepsilon_{pré} E_p A_p (d_p - y_{cg})}{\frac{1}{2\alpha_e} \frac{x}{d-x} \left(x b_f (y_{cg} - \frac{x}{3}) - \frac{(x-h_f)^2}{x} (b_f - b_w) (y_{cg} - \frac{x}{3} + \frac{h_f}{3}) \right) + A_s' \frac{x-d'}{d-x} (y_{cg} - d') + A_s (d - y_{cg}) + \left(A_p \frac{E_p}{E_s} \frac{d_p - x}{d-x} \right) (d_p - y_{cg})} \quad (2.140)$$

Para protensão não aderente, não há compatibilidade de deformações entre a armadura ativa e a seção de concreto, portanto, a parcela relativa à $\Delta\varepsilon_p$ é considerada nula, conforme (2.141).

$$\sigma_s = \frac{M_{CF} - \varepsilon_{pré} E_p A_p (d_p - y_{cg})}{\frac{1}{2\alpha_e} \frac{x}{d-x} \left(x b_f (y_{cg} - \frac{x}{3}) - \frac{(x-h_f)^2}{x} (b_f - b_w) (y_{cg} - \frac{x}{3} + \frac{h_f}{3}) \right) + A_s' \frac{x-d'}{d-x} (y_{cg} - d') + A_s (d - y_{cg})} \quad (2.141)$$

Assim, têm-se pares de duas equações: (2.136) e (2.140) para protensão aderente em seções T com linha neutra na alma, e (2.137) e (2.141) para protensão não aderente, e duas incógnitas σ_s e x . Portanto, são conhecidos os dados para determinação do valor característico da abertura das fissuras.

O cálculo da posição da linha neutra e tensão na armadura no estágio II para peças protendidas gera discussões. França (2001) calcula a posição da linha neutra para seções retangulares considerando o equilíbrio da seção, escreve as deformações em função da curvatura. Calcula também a posição da linha neutra para seções T, neste caso, desenvolve os cálculos com as deformações dos materiais em função da deformação do concreto, e utiliza os esforços solicitantes equivalentes como dados de entrada para o problema. França (2001) explica que para protensão não aderente, deve-se adotar $\Delta\varepsilon_p \cong 0$.

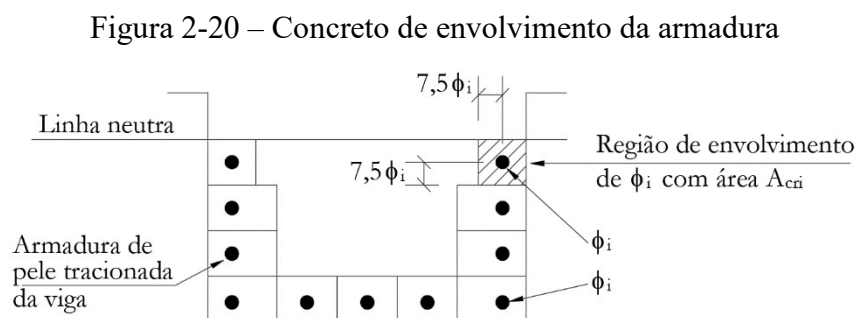
Cholfe e Bonilha (2013) apresentam o cálculo da posição da linha neutra e tensão na armadura no estágio II para seções retangulares, sem armadura dupla e considerando protensão não aderente, utilizam o equilíbrio da seção.

Já Zanette (2006), apresenta o cálculo de tensão na armadura no estágio II em casos de protensão não aderente para seções retangulares. Inicialmente, considera a seção homogeneizada no estágio II como se a estrutura fosse de concreto armado e calcula o momento de inércia I_{II} , a posição da linha neutra x e a área da seção homogeneizada A_{II} . Então, considera o efeito da protensão como um carregamento externo equivalente, decomposto em um esforço normal, aplicado no centro de gravidade da seção bruta de concreto, e um momento fletor considerado nas combinações de serviço.

Nilson (1976) calcula as tensões nos materiais e posição da linha neutra para uma viga parcialmente protendida aderente. Explica que para vigas de concreto armado, o cálculo das tensões na seção fissurada é simples, o conceito de seção homogeneizada permite o uso de equações da mecânica para o cálculo da posição da linha neutra. Porém, para vigas parcialmente protendidas, a posição da linha neutra no estágio II não depende apenas da geometria da seção transversal e das propriedades dos materiais, mas também da força de protensão e do carregamento. Nilson (1976) realiza o cálculo da posição da linha neutra utilizando o equilíbrio de momentos na seção homogeneizada.

Baseando-se em França (2001) e Cholfe e Bonilha (2013), neste trabalho, foram desenvolvidas equações de equilíbrio de forças e momentos na seção transversal no estágio II para a combinação frequente. Assim, foram obtidos pares de equações para as tensões nas armaduras passivas em função da posição da linha neutra, conforme as equações (2.107) a (2.141), apresentadas anteriormente. A posição da linha neutra e tensão na armadura passiva são, então, determinadas por processo iterativo.

Conforme a NBR 6118:2014, para cada elemento ou grupo de elementos de armaduras passiva e ativa aderente que controla a fissuração, excluídos cabos em bainhas, considera-se uma área da região de envolvimento do concreto, constituída por retângulos com lados à distância máxima de $7,5 \phi$ do eixo de cada elemento de armadura, ver Figura 2-20.



Fonte: Adaptada da NBR 6118:2014.

O valor característico da abertura de fissuras, calculado para cada parte da região de envolvimento, deve ser o menor dos obtidos nas equações (2.142) e (2.143)

$$w_k = \frac{\phi_i}{12,5\eta_1} \frac{\sigma_{si}}{E_{si}} \frac{3\sigma_{si}}{f_{ctm}} \quad (2.142)$$

$$w_k = \frac{\phi_i}{12,5\eta_1} \frac{\sigma_{si}}{E_{si}} \left(\frac{4}{\rho_{ri}} + 45 \right) \quad (2.143)$$

em que

σ_{si} , ϕ_i , E_{si} e ρ_{ri} são definidos para cada área de envolvimento em exame;

A_{cri} é a área da região de envolvimento protegida pela barra ϕ_i ;

E_{si} é o módulo de elasticidade do aço da barra considerada;

ϕ_i é o diâmetro da barra que protege a região de envolvimento considerada;

ρ_{ri} é a taxa de armadura passiva ou ativa aderente, exceto em bainhas, em relação à área da região de envolvimento A_{cri} ;

σ_{si} é o acréscimo de tensão entre o estado-limite de descompressão e o carregamento frequente, no centro de gravidade da armadura considerada; que deve ser calculado no estágio II considerando toda a armadura ativa, inclusive aquela dentro de bainhas;

η_1 é o coeficiente de conformação superficial da armadura passiva considerada, devendo ser substituído por η_{p1} para armaduras ativas, conforme valores abaixo:

$\eta_1 = 1,0$ para barras lisas;

$\eta_1 = 1,4$ para barras entalhadas;

$\eta_1 = 2,25$ para barras nervuradas;

$\eta_{p1} = 1,0$ para fios lisos;

$\eta_{p1} = 1,2$ para cordoalhas de três a sete fios;

$\eta_{p1} = 1,4$ para fios dentados.

2.1.10 Verificação do estado-limite de deformações de excessivas

Conforme a NBR 6118:2014, o estado-limite de deformações excessivas é aquele em que as deformações atingem os limites estabelecidos para a utilização normal. Os deslocamentos-limites são classificados nos grupos abaixo:

- a) aceitabilidade sensorial: caracterizado por vibrações indesejáveis ou efeitos visuais desagradáveis;
- b) efeitos específicos: os deslocamentos podem impedir a utilização adequada da construção;
- c) efeitos em elementos não estruturais: deslocamentos estruturais podem provocar mau funcionamento de elementos ligados a estrutura;

- d) efeitos em elementos estruturais: os deslocamentos podem afetar o comportamento de um elemento estrutural, se os deslocamentos forem relevantes para o elemento considerado, devem ser considerados seus efeitos sobre as tensões ou estabilidade da estrutura.

Os Quadros 2.11 e 2.12 indicam os valores dos deslocamentos-limites para estruturas em serviço.

Quadro 2.11 – Limites para deslocamento

Tipo de efeito	Razão da limitação	Exemplo	Deslocamento a considerar	Deslocamento-limite
Aceitabilidade sensorial	Visual	Deslocamentos visíveis em elementos estruturais	Total	$l/250$
	Outro	Vibrações sentidas no piso	Devido a cargas acidentais	$l/350$
Efeitos estruturais em serviço	Superfícies que devem drenar água	Coberturas e varandas	Total	$l/250^a$
	Pavimentos que devem permanecer planos	Ginásios e pistas de boliche	Total	$l/350 + \text{contraflecha}^b$
			Ocorrido após a construção do piso	$l/600$
	Elementos que suportam equipamentos sensíveis	Laboratórios	Ocorrido após nivelamento do equipamento	De acordo com recomendação do fabricante do equipamento
Efeitos em elementos não estruturais	Paredes	Alvenaria, caixilhos e revestimentos	Após a construção da parede	$l/500^c$ e 10 mm e $\theta = 0,0017 \text{ rad}^d$
		Divisórias leves e caixilhos telescópicos	Ocorrido após a instalação da divisória	$l/250^c$ e 25 mm
		Movimento lateral de edifícios	Provocado pela ação do vento para combinação frequente ($\psi_1 = 0,30$)	$H/1700$ e $H/850^e$ entre pavimentos ^f
		Movimentos térmicos verticais	Provocado por diferença de temperatura	$l/400^g$ e 15 mm

Fonte: NBR 6118:2014.

Quadro 2.12 – Limites para deslocamento (continuação)

Tipo de efeito	Razão da limitação	Exemplo	Deslocamento a considerar	Deslocamento-limite
Efeitos em elementos não estruturais	Forros	Movimentos térmicos horizontais	Provocado por diferença de temperatura	$H_i/500$
		Revestimentos colados	Ocorrido após a construção do forro	$\ell/350$
		Revestimentos pendurados ou com juntas	Deslocamento ocorrido após a construção do forro	$\ell/175$
	Pontes rolantes	Desalinhamento de trilhos	Deslocamento provocado pelas ações decorrentes da frenagem	$H/400$
Efeitos em elementos estruturais	Afastamento em relação às hipóteses de cálculo adotadas	Se os deslocamentos forem relevantes para o elemento considerado, seus efeitos sobre as tensões ou sobre a estabilidade da estrutura devem ser considerados, incorporando-os ao modelo estrutural adotado.		

a As superfícies devem ser suficientemente inclinadas ou o deslocamento previsto compensado por contraflechas, de modo a não se ter acúmulo de água.

b Os deslocamentos podem ser parcialmente compensados pela especificação de contraflechas. Entretanto, a atuação isolada da contraflecha não pode ocasionar um desvio do plano maior que $\ell/350$.

c O vão ℓ deve ser tomado na direção na qual a parede ou a divisória se desenvolve.

d Rotação nos elementos que suportam paredes.

e H é a altura total do edifício e H_i o desnível entre dois pavimentos vizinhos.

f Esse limite aplica-se ao deslocamento lateral entre dois pavimentos consecutivos, devido à atuação de ações horizontais. Não podem ser incluídos os deslocamentos devidos a deformações axiais nos pilares. O limite também se aplica ao deslocamento vertical relativo das extremidades de lintéis conectados a duas paredes de contraventamento, quando H_i representa o comprimento do lintel.

g O valor ℓ refere-se à distância entre o pilar externo e o primeiro pilar interno.

NOTAS

1 Todos os valores-limites de deslocamentos supõem elementos de vão ℓ suportados em ambas as extremidades por apoios que não se movem. Quando se tratar de balanços, o vão equivalente a ser considerado deve ser o dobro do comprimento do balanço.

2 Para o caso de elementos de superfície, os limites prescritos consideram que o valor ℓ é o menor vão, exceto em casos de verificação de paredes e divisórias, onde interessa a direção na qual a parede ou divisória se desenvolve, limitando-se esse valor a duas vezes o vão menor.

3 O deslocamento total deve ser obtido a partir da combinação das ações características ponderadas pelos coeficientes definidos na Seção 11.

4 Deslocamentos excessivos podem ser parcialmente compensados por contraflechas.

Fonte: NBR 6118:2014.

A NBR 6118:2014 estabelece que a verificação dos valores limites de deslocamentos para elementos estruturais lineares deve ser realizada na combinação quase permanente, considerando a rigidez efetiva das seções do elemento estrutural, ou seja, considerando a presença da armadura, a existência de fissuras no concreto ao longo da armadura e as deformações diferidas.

Entre outras coisas, salienta-se que a deformação real da estrutura depende do dimensionamento, do processo construtivo, das propriedades dos materiais no momento da solicitação, e com a grande variabilidade desses parâmetros, não há grande precisão na previsão dos deslocamentos.

Admite-se o comportamento elástico e linear do concreto e aço, desde que as tensões solicitantes não ultrapassem o estado-limite de formação de fissuras, assim as seções ao longo do elemento estrutural podem ter as deformações específicas determinadas no estágio I. Assim, a rigidez equivalente $(EI)_{eq}$ da peça é dada pela equação (2.144),

$$(EI)_{eq} = E_{cs}I_c \quad (2.144)$$

em que E_{cs} é o módulo de elasticidade secante do concreto e I_c é o momento de inércia da seção bruta de concreto.

Em caso contrário, deve-se realizar a análise no estágio II, assim, a rigidez equivalente $(EI)_{eq}$ da peça é dada pela equação (2.145),

$$(EI)_{eq} = E_{cs} \left\{ \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 I_c + \left[1 - \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 \right] I_{II} \right\} \leq E_{cs}I_c \quad (2.145)$$

em que M_{CQP} é o momento fletor, na combinação quase permanente, na seção crítica do vão considerado, e M_r é o momento de fissuração, dado por (2.146). Observa-se que o momento de fissuração para elementos protendidos difere dos elementos em concreto armado pela consideração do momento de descompressão M_0 , apresentado em (2.147).

$$M_r = M_0 + W_c f_{ct} \quad (2.146)$$

$$M_0 = P_{inf} \left(e + \frac{W_c}{A_c} \right) \quad (2.147)$$

P_{inf} é a força de protensão após as perdas, e é excentricidade do cabo equivalente, W_c é o módulo de resistência da seção, A_c é área da seção bruta de concreto e f_{ct} é dado por (2.148),

$$f_{ct} = \alpha f_{ctk,inf} \quad (2.148)$$

sendo α igual a 1,2 para seções T e igual a 1,5 para seções retangulares, e $f_{ctk,inf}$ dado por (2.149).

$$f_{ctk,inf} = 0,7 \times 0,3 \times f_{ck}^{2/3} \quad (2.149)$$

Finalmente, I_{II} é o momento de inércia da seção fissurada de concreto no estágio II. A posição da linha neutra x é calculada de maneira análoga à apresentada no item 2.1.9, porém, utiliza-se o momento da combinação quase permanente para sua determinação.

Para seções retangulares ou T com linha neutra na mesa, e protensão aderente, I_{II} é obtido pela expressão (2.159), em que b_c é tomado como b_w ou b_f , conforme o caso,

$$I_{II} = \frac{b_c x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \left(\frac{E_s}{E_{cs}} - 1 \right) A_s' (x - d')^2 + \frac{E_p}{E_{cs}} A_p (d_p - x)^2 \quad (2.150)$$

Para protensão não aderente, desconsidera-se a armadura ativa no cálculo de I_{II} , ver (2.151),

$$I_{II} = \frac{b_c x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \left(\frac{E_s}{E_{cs}} - 1 \right) A_s' (x - d')^2 \quad (2.151)$$

Já para seções T com linha neutra na mesa, e protensão aderente, I_{II} é obtido pela expressão (2.152),

$$I_{II} = \frac{(b_f - b_w) h_f^3}{12} + (b_f - b_w) h_f \left(x - h_f/2 \right)^2 + \frac{b_w x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \left(\frac{E_s}{E_{cs}} - 1 \right) A_s' (x - d')^2 + \frac{E_p}{E_{cs}} A_p (d_p - x)^2 \quad (2.152)$$

Para protensão não aderente, desconsidera-se a armadura ativa no cálculo de I_{II} , ver (2.153).

$$I_{II} = \frac{(b_f - b_w) h_f^3}{12} + (b_f - b_w) h_f \left(x - h_f/2 \right)^2 + \frac{b_w x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \left(\frac{E_s}{E_{cs}} - 1 \right) A_s' (x - d')^2 \quad (2.153)$$

Pela equação da linha elástica, são obtidos os deslocamentos ao longo da viga, ver (2.154) são considerados os momentos solicitantes na combinação quase permanente e momento devido à carga equivalente de alívio da protensão. Nos casos em que as tensões atuantes ultrapassam o estado-limite de formação de fissuras, esses deslocamentos devem ser modificados conforme a expressão (2.155).

$$f_{elástica} = \iint - \frac{M}{E_{cs} I_c} \quad (2.154)$$

$$f_{mediata} = \frac{E_{cs} I_c}{(EI)_{eq}} f_{elástica} \quad (2.155)$$

Após o cálculo das flechas imediatas, calcula-se a flecha diferida, devida ao carregamento permanente, pela equação (2.156), em que φ_∞ é o coeficiente de fluência.

$$f_{diferida,g} = (1 + \varphi_\infty) f_{mediata,g} \quad (2.156)$$

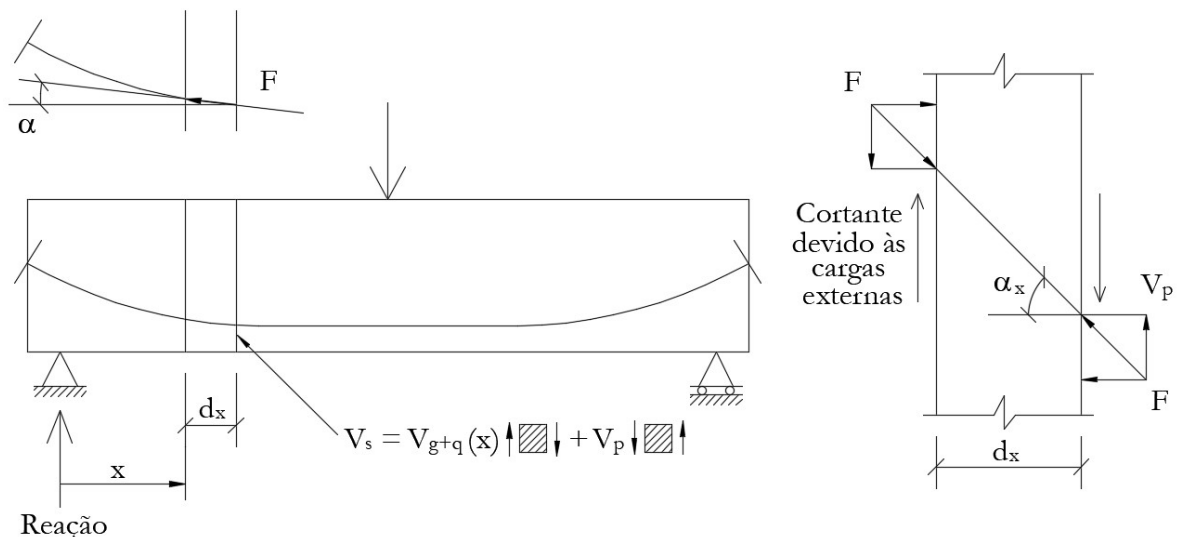
Assim, pode ser obtida a flecha total pela soma da flecha diferida com a flecha imediata devida ao carregamento acidental, ver (2.157) e, então, comparada à flecha admissível.

$$f_{total} = f_{diferida,g} + f_{mediata,q} \quad (2.157)$$

2.1.11 Dimensionamento à força cortante

O concreto protendido apresenta duas vantagens em relação ao concreto armado, no que diz respeito à resistência à força cortante. A primeira é que, para o mesmo carregamento e condições, o esforço cortante no concreto protendido é menor, devido à inclinação da força de protensão que gera uma componente vertical geralmente oposta ao carregamento externo, como se pode ver na Figura 2-21. A segunda é a redução da tração diagonal devido à compressão induzida pela protensão. O seu ângulo de inclinação em relação ao eixo longitudinal da peça é reduzido.

Figura 2-21 – Redução da força cortante no caso de cabo inclinado



Fonte: Adaptada de Naaman (2012).

A NBR 6118:2014 apresenta prescrições para o dimensionamento de elementos lineares armados ou protendidos, submetidos a esforços cortantes, eventualmente combinados com outras solicitações.

No valor da força cortante solicitante de cálculo, deve ser considerado o efeito da força de protensão na sua direção, $V_{p\infty}$, com o valor de cálculo correspondente ao tempo t considerado. Quando esse efeito for favorável, a armadura longitudinal de tração junto à face tracionada por flexão deve satisfazer a expressão (2.158)

$$A_p f_{pyd} + A_s f_{yd} \geq V_{sd} \quad (2.158)$$

Assim, a força cortante solicitante de cálculo é obtida por (2.159)

$$V_{Sd} = \gamma_{fg}V_g + \gamma_{fq}V_q + \gamma_{fp}V_{P\infty} \quad (2.159)$$

em que γ_f é o coeficiente de ponderação das ações obtido pelo Quadro 2.13.

Quadro 2.13 – Coeficiente $\gamma_f = \gamma_{f1} \cdot \gamma_{f3}$

Combinações de ações	Ações							
	Permanentes (g)		Variáveis (q)		Protensão (p)		Recalques de apoio e retração	
	D	F	G	T	D	F	D	F
Normais	1,4 ^a	1,0	1,4	1,2	1,2	0,9	1,2	0
Especiais ou de construção	1,3	1,0	1,2	1,0	1,2	0,9	1,2	0
Excepcionais	1,2	1,0	1,0	0	1,2	0,9	0	0

onde
D é desfavorável, *F* é favorável, *G* representa as cargas variáveis em geral e *T* é a temperatura.
^a Para as cargas permanentes de pequena variabilidade, como o peso próprio das estruturas, especialmente as pré-moldadas, esse coeficiente pode ser reduzido para 1,3.

Fonte: NBR 6118:2014.

No dimensionamento de vigas, devem ser satisfeitas, simultaneamente, as expressões (2.160) e (2.161)

$$V_{Sd} \leq V_{Rd2} \quad (2.160)$$

$$V_{Sd} \leq V_{Rd3} = V_c + V_{sw} \quad (2.161)$$

em que

V_{Rd2} é a força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas de concreto;

V_{Rd3} é a força cortante resistente de cálculo relativa à ruína por tração diagonal;

V_c é a parcela de força cortante absorvida por mecanismos complementares ao da treliça;

V_{sw} é a parcela resistida pela armadura transversal.

O modelo de cálculo I admite diagonais de compressão inclinadas de 45° em relação ao eixo longitudinal do elemento estrutural e, também, que V_c tem valor constante, independentemente de V_{Sd} . Enquanto o modelo de cálculo II admite diagonais de compressão

com inclinação em relação ao eixo longitudinal variando entre 30° a 45°, e que V_c tem redução com o aumento da força cortante solicitante de cálculo.

Para o modelo de cálculo I, a força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas de concreto é calculada por (2.162),

$$V_{Rd2} = 0,27\alpha_{v2}f_{cd}b_wd \quad (2.162)$$

em que α_{v2} é dado por (2.163), com f_{ck} em Mpa;

$$\alpha_{v2} = 1 - \frac{f_{ck}}{250} \quad (2.163)$$

b_w é a menor largura da seção, caso haja bainhas injetadas com diâmetro superior a $b_w/8$, a largura resistente deve ser obtida por (2.164), na posição da alma em que esse valor seja mais desfavorável, com exceção do nível que define o banzo tracionado da viga;

$$b_{w,ef} = b_w - \frac{1}{2} \sum \phi \quad (2.164)$$

d é a altura útil da seção, igual à distância da borda comprimida ao centro de gravidade da armadura de tração, para vigas protendidas com cabos distribuídos ao longo da altura, não há necessidade de adotar valor inferior 0,8 h, desde que exista armadura junto à face tracionada e seja satisfeita a expressão (2.158).

Já para o modelo de cálculo II, a força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas de concreto é calculada por (2.165), em que α é o ângulo da diagonal tracionada em relação ao eixo longitudinal e θ é o ângulo da diagonal comprimida em relação ao eixo longitudinal.

$$V_{Rd2} = 0,54\alpha_{v2}f_{cd}b_wd\text{sen}^2\theta(\text{cota} + \text{cot}\theta) \quad (2.165)$$

Retornando à força cortante resistente de cálculo relativa à ruína por tração diagonal, essa é obtida por (2.166)

$$V_{Rd3} = V_c + V_{sw} \quad (2.166)$$

em que V_{sw} é calculada pela equação (2.167)

$$V_{sw} = \left(\frac{A_{sw}}{s}\right) 0,9 d f_{ywd}(\text{sen} \alpha + \text{cos} \alpha) \quad (2.167)$$

sendo

A_{sw} a área da seção de aço da armadura transversal;

s o espaçamento entre as barras da armadura transversal, no sentido longitudinal da peça;

f_{ywd} a tensão na armadura transversal passiva;

α o ângulo de inclinação da armadura transversal em relação ao eixo longitudinal do elemento.

A parcela de força cortante absorvida por mecanismos complementares ao da treliça V_c é igual a zero nos elementos estruturais tracionados quando a linha neutra se encontra fora da seção.

No modelo de cálculo I, em caso de flexão simples ou na flexo-tração com a linha neutra cortando a seção, V_c é igual a V_{c0} , dado por (2.168), e para flexo-compressão, V_c é calculado por (2.169)

$$V_{c0} = 0,6f_{ctd}b_w d \quad (2.168)$$

$$V_c = V_{c0} \left(1 + \frac{M_0}{M_{Sd,máx}} \right) \leq 2V_{c0} \quad (2.169)$$

sendo M_0 o valor do momento fletor que anula a tensão normal de compressão na borda tracionada por $M_{Sd,máx}$, são considerados somente os isostáticos de protensão, com o coeficiente de ponderação das cargas devidas à protensão igual a 0,9; e $M_{Sd,máx}$ é o momento fletor de cálculo máximo no trecho em análise, são considerados apenas os hiperestáticos de protensão.

No modelo de cálculo II, em caso de flexão simples ou na flexo-tração com a linha neutra cortando a seção, V_c é igual a V_{c1} , e para flexo-compressão, V_c é calculado por (2.170). Sendo que V_{c1} é igual a V_{c0} , quando $V_{Sd} \leq V_{c0}$; V_{c1} é igual a zero, quando $V_{Sd} = V_{Rd2}$; interpolando-se linearmente para valores intermediários.

$$V_c = V_{c1} \left(1 + \frac{M_0}{M_{Sd,máx}} \right) \leq 2V_{c1} \quad (2.170)$$

Assim, para estribos a 90°, a área de aço da armadura transversal é dada por (2.171)

$$\left(\frac{A_{sw}}{s} \right) = \frac{V_{Sd} - V_c}{0,9 d f_{ywd}} \quad (2.171)$$

Conforme a NBR 6118:2014, em casos de cargas próximas aos apoios, para o cálculo da armadura transversal, em caso de apoio direto (se a carga e a reação de apoio forem aplicadas em faces opostas do elemento estrutural, comprimindo-o), vale:

- a) no trecho entre o apoio e a seção situada à distância $d/2$ da face de apoio, a força cortante devida a cargas distribuídas pode ser considerada constante e igual à dessa seção;
- b) a força cortante devida a cargas concentradas aplicada a uma distância $a \leq 2d$ do eixo teórico do apoio pode, nesse trecho, ser reduzida, multiplicando-a por $a/(2d)$. Essa redução não é aplicada a forças cortantes devidas dos cabos inclinados de protensão.

As reduções acima não se aplicam à verificação da biela comprimida e nem a apoios indiretos.

3 PROGRAMA COMPUTACIONAL

Dominicini e Coelho (2014) implementaram uma ferramenta computacional para dimensionamento de vigas isostáticas protendidas à flexão, para protensão completa e limitada, conforme a NBR 6118:2007, porém, não se contemplava o nível de protensão parcial. Neste trabalho, realizou-se a adaptação do programa à NBR 6118: 2014, a expansão do programa, incluindo a protensão parcial, a verificação do estado-limite de deformação excessiva e o dimensionamento à força cortante.

Trata-se de uma ferramenta computacional acadêmica, desenvolvida utilizando o *Microsoft Excel* e o *Microsoft Visual Basic for Applications*, com interface gráfica e geração de relatórios de cálculo, que seja utilizada como recurso auxiliar na disciplina Concreto Protendido, tanto no nível de graduação e quanto de pós-graduação. O Apêndice 1 apresenta o código do programa.

A ferramenta desenvolvida apresenta tela inicial com uma barra de ferramentas, com ícones para entrada de dados e geração de memórias de cálculo. Inicialmente, o usuário deve acessar a janela “Propriedades Geométricas da Seção / Concreto”, pelo primeiro ícone da barra de ferramentas, ver Figura 3-1 e Figura 3-2.

Figura 3-1 – Primeiro ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-2 – Propriedades geométricas da seção / concreto

Propriedades Geométricas da Seção / Concreto

Propriedades Geométricas

Inserir dados de seção já conhecida

ID	B	H	y	Seção
1	0,2	0,85	0,425	Ret

Adicionar Remover Limpar

Altura (H) da seção: m

Vão da viga: m

nº de seções: (mínimo de 10 seções)

Concreto

fck: MPa Eci

fckj: MPa Ecs

Aço

Ep: MPa

Seguir roteiro automático

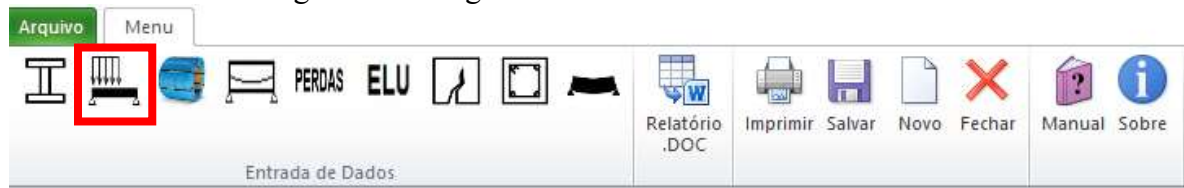
OK

Fonte: Elaborada pela autora.

O usuário deve definir a geometria da seção transversal, T ou retangular, o vão da viga e o número de seções de análise a serem realizadas pelo programa. Então, deve-se informar a resistência característica à compressão do concreto, a resistência à compressão do concreto no momento da protensão e o módulo de elasticidade da armadura ativa. Salienta-se que o programa é limitado à classe C50. A seção transversal da viga deve ser informada como composição de seções elementares, para cálculo de suas propriedades geométricas, ou então, podem-se fornecer as propriedades já conhecidas da seção.

Em seguida, o usuário deve acessar a janela “Nível de Protensão / Cargas”, ver Figura 3-3 e Figura 3-4.

Figura 3-3 – Segundo ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-4 – Nível de protensão / cargas

Nível de Protensão / Cargas

Nível de Protensão

Classes de Agressividade Ambiental (CAA)

I - Fraca

II - Moderada

III - Forte

IV - Muito Forte

Tipo de Protensão

Pós-Tração

Nível de Protensão:

Alterar Nível de Protensão manualmente

Cargas

ID	TIPO	VALOR	X_i	X_f	CATEGORIA	Psi0	Psi1	Psi2
1	Dist.	10,75	0	12	Perm	-	-	-
2	Dist.	5	0	12	Var	0,5	0,4	0,3

Remover Carga

Concentrada

Permanente

Distribuída uniforme

Variável

Adicionar Carga

Excentricidade para Pré-Dimensionamento

ycg: 0,425 m ep: m

Tensão no ELS-F

1,5 x ftk (Retang.) 1,2 x ftk (T)

Perda de Carga Estimada

%

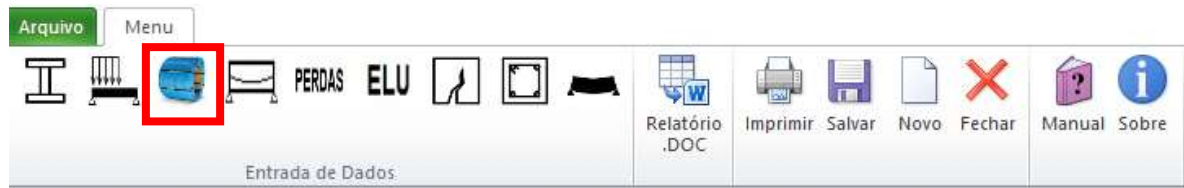
OK

Fonte: Elaborada pela autora.

Define-se a classe de agressividade ambiental e nível de protensão. No caso de protensão parcial, informa-se a porcentagem do momento permanente a balancear, para cálculo da área de armadura ativa necessária. Então, o usuário define os carregamentos permanentes e acidentais, distribuídos e ou concentrados. Além disso, deve informar a excentricidade e porcentagem estimada de perda para o pré-dimensionamento da força de protensão inicial. O programa calcula o momento fletor solicitante e a força cortante solicitante.

Continuando, usuário deve acessar a janela “Aço / Fuso limite”, ver Figura 3-5 e Figura 3-6.

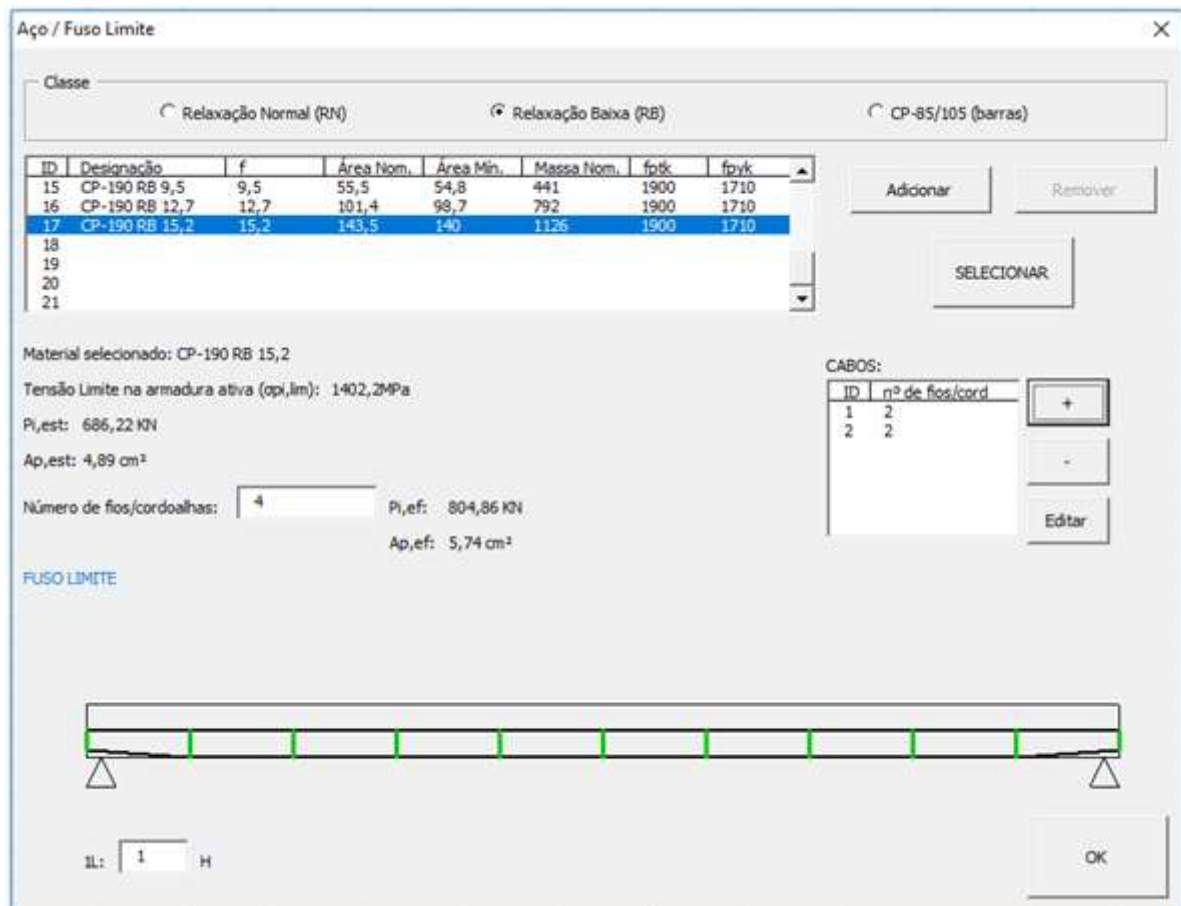
Figura 3-5 – Terceiro ícone da barra de ferramentas



Fonte: Elaborada pela autora.

O programa calcula a força de protensão inicial necessária e fornece opções para a armadura ativa a ser adotada. Realizada a escolha da armadura ativa pelo usuário, a ferramenta calcula o número necessário de fios, cordoalhas ou barras. O usuário deve definir em quantos cabos estes elementos devem ser alojados. O programa fornece o fuso limite para os cabos.

Figura 3-6 – Aço / Fuso limite



Fonte: Elaborada pela autora.

É gerado um relatório geral da viga, contendo informações das propriedades geométricas da seção, vão da viga, propriedades dos materiais e dados de pré-dimensionamento.

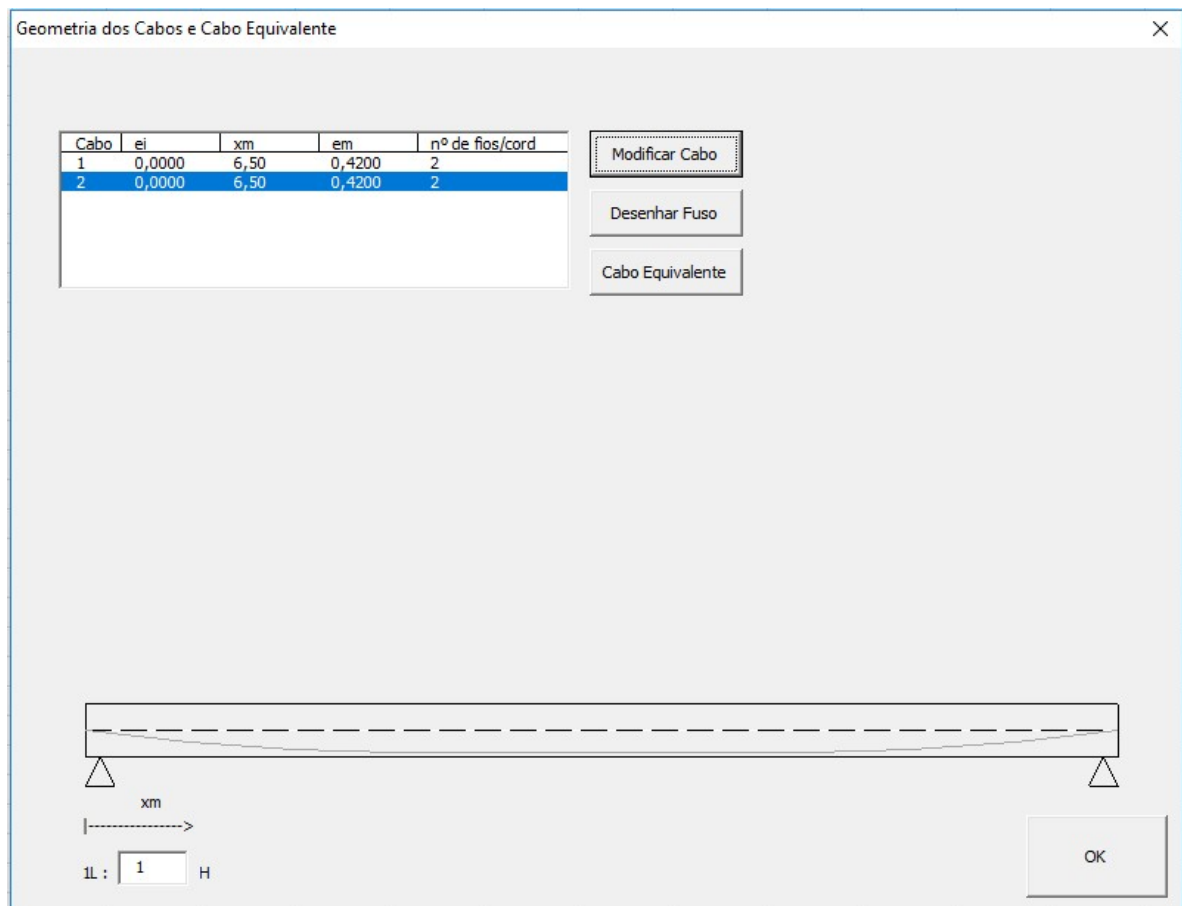
Na sequência, o usuário deve acessar a janela “Geometria dos Cabos e Cabo Equivalente” e informar a excentricidade inicial dos cabos, o comprimento dos trechos parabólicos e as excentricidades nos trechos retos. É possível adotar, também, cabos completamente parabólicos ou retilíneos. Ver Figura 3-7 e Figura 3-8.

Figura 3-7 – Quarto ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-8 – Geometria dos cabos e cabo equivalente

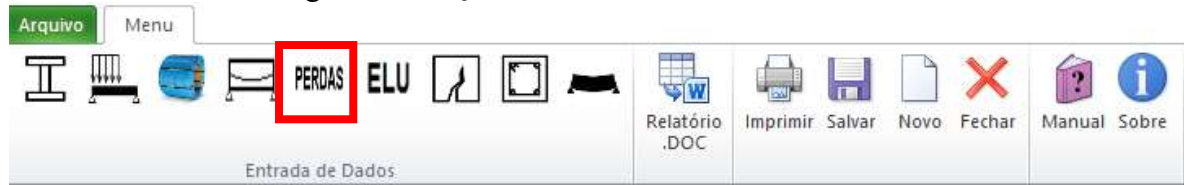


Fonte: Elaborada pela autora.

Em seguida, o usuário deve acessar a janela “Perdas Imediatas e Progressivas”, ver Figura 3-9 e Figura 3-10. Deve informar o tipo de superfície de atrito cabo-bainha, o desvio parasitário, o

tipo de ancoragem, o encurtamento do cabo pela acomodação da ancoragem e os dados referentes às perdas progressivas. O programa calcula as perdas de protensão imediatas e progressivas. Calcula, também, a força de protensão ao longo do cabo e tensões nas seções de análise após as perdas, verifica os estados-limites de serviço conforme cada nível de protensão. Para protensão parcial, nas seções que ultrapassam o estado-limite de formação de fissuras na combinação frequente, emite-se um aviso indicando a verificação do estado-limite de abertura das fissuras, que é analisado pelo programa posteriormente. Ver Figura 3-9 e Figura 3-10. É gerado um relatório com os valores da força inicial e final de protensão, das perdas, da porcentagem de perdas e das tensões nas seções de análise.

Figura 3-9 – Quinto ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-10 – Perdas imediatas e progressivas

Perdas Imediatas e Progressivas

PERDA POR ATRITO CABO-BAINHA (Pós-Tração)

TIPOS DE SUPERFÍCIE DE ATRITO:

- Entre cabo e concreto
- Entre barras ou fios com mossas ou saliências e bainha metálica
- Entre fios lisos ou cordoalhas e bainha metálica
- Entre fios lisos ou cordoalhas e bainha metálica lubrificada
- Entre cordoalha e bainha de polipropileno lubrificada

DESVIO PARASITÁRIO (k)

- 0,01.μ
- Outro

PERDA POR DEFORMAÇÃO DA ANCORAGEM

ΔL mm

PERDA POR ENCURTAMENTO ELÁSTICO

Não é necessário entrada de dados.

PERDAS PROGRESSIVAS (Processo Simplificado)

Umidade relativa do ar: %

Temperatura média: °C

Slump: cm

t₀: dias

Perímetro em contato com o ar: cm

Tipo de cimento

- CP III e IV
- CP I e II
- CP V-ARI

ϕ_{∞}

ϵ_{cs}

ANCORAGEM

- Ativa - Ativa
- Ativa - Passiva

OK

Fonte: Elaborada pela autora.

Para o dimensionamento no estado-limite último, o usuário deve acessar a janela “Estado limite último” e informar a largura da alma, dimensões da mesa, altura útil das armaduras passiva e ativa. Então, o programa realiza o dimensionamento e gera um relatório com a área de aço necessária em cada seção de análise (não é considerada a decalagem). Ver Figura 3-11 e Figura 3-12. É gerado um relatório com o momento fletor solicitante de cálculo, força de protensão após as perdas, razão entre a posição da linha neutra e altura útil da seção, área de armadura passiva longitudinal necessária e mínima para as seções de análise.

Figura 3-11 – Sexto ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-12 – Estado-limite último

Fonte: Elaborada pela autora.

Para protensão parcial, após a obtenção da área de armadura passiva necessária, realiza-se a verificação do estado-limite de abertura das fissuras na seção mais solicitada, ver Figura 3-13 e Figura 3-14.

Figura 3-13 – Sétimo ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-14 – Verificação do estado-limite de abertura das fissuras

Verificação da abertura de fissuras

Armadura passiva na seção crítica

Seção crítica: $x = 9,65$ m

As, nec: 17,245 cm²

Bitola longitudinal: 20

Bitola estribo: 5

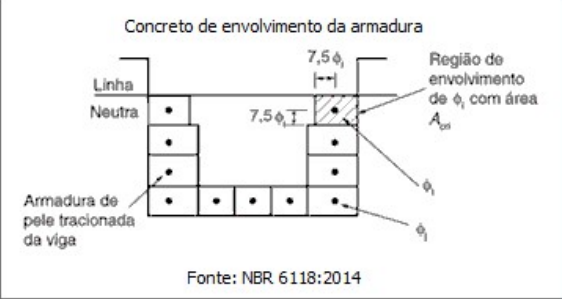
Nº ramos: 2

Diâmetro do Agregado: 19 mm

Cobrimento: 2,5 cm

Armação: 6 ϕ 20 As, adot: 18,85 cm²

Alojamento:
1 camadas de 6 ϕ 20



Fonte: NBR 6118:2014

Verificação da abertura de fissuras na seção central do vão

Acri: 570 cm²

Tensão de tração: $\sigma_s = 13,70$ kN/cm²

wk: 0,066 mm

Verificação: Ok!

Fonte: Elaborada pela autora.

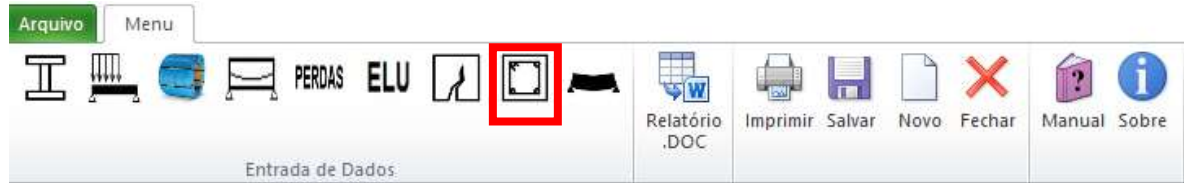
Inicialmente, o usuário deve definir o diâmetro da armadura passiva longitudinal e da armadura transversal, bem como o número de ramos dos estribos, diâmetro do agregado e cobrimento das armaduras passivas. A partir destes dados, o programa aloja a armadura longitudinal e indica quantas camadas serão utilizadas e o número de barras por camada, assim, calcula e fornece a área da região de envolvimento da armadura.

Em seguida, para a seção mais solicitada, o programa calcula a tensão na armadura passiva, σ_s , e a posição da linha neutra x . Então, é calculado automaticamente o valor da abertura das fissuras e comparado com o valor máximo de 0,2 mm. É gerado um relatório informando a posição da linha neutra no estágio II, o valor da tensão na armadura passiva, a área da região de envolvimento das armaduras e o valor característico da abertura das fissuras, indicando se foi atendido o estado-limite de abertura das fissuras.

Prosseguindo, realiza-se o dimensionamento à força cortante. Ver Figura 3-15 e Figura 3-16. O usuário deve arbitrar três faixas para a análise e distribuição dos estribos, conforme a Figura 3-16 e Figura 3-17. A primeira vem desde o primeiro apoio e se estende até a coordenada final desta faixa, x_{f1} , que deve ser fornecida pelo usuário. A segunda faixa tem

início a partir da primeira, portanto, em $x_{i2} = x_{f1}$, deve ser indicada a sua coordenada final x_{f2} . A terceira faixa inicia-se na coordenada em $x_{i3} = x_{f2}$, e se estende até o segundo apoio.

Figura 3-15 – Oitavo ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-16 – Dimensionamento à força cortante

Dimensionamento à força cortante

Modelo de cálculo

Modelo1

Modelo2

Verificação da biela comprimida e armadura mínima

ϕ bainha: 15,2 mm Aço estribo: CA60

Redução VSd

bw,efet: 30,00 cm Asw,min: 3,48 cm²/m

Vsd,máx: 210,55 kN VSd,Asmín: 453,50 kN

VRd2: 1466,3 kN Biela comprimida OK

Vc: 322,93 kN

Armação transversal - 1ª faixa

Comprimento da faixa: 650 cm

xi1: 0 cm

xf1: 650 cm

Vsd,máx,faixa1: 210,5533 kN

Asw,nec: -2,9917 cm²/m

Nº ramos: 2

Bitola: 5 mm

Alojamento: 60 ϕ 5 c/ 11 (2 R)

Asw,adot: 3,62 cm²/m

Armação transversal - 2ª faixa

Comprimento da faixa: 630 cm

xi2: 650 cm

xf2: 1280 cm

Vsd,máx,faixa2: 68,72985 kN

Asw,nec: -6,7671 cm²/m

Nº ramos: 2

Bitola: 5 mm

Alojamento: 58 ϕ 5 c/ 11 (2 R)

Asw,adot: 3,61 cm²/m

Armação transversal - 3ª faixa

Comprimento da faixa: 650 cm

xi3: 1280 cm

xf3: 1930 cm

Vsd,máx,faixa3: 210,5533 kN

Asw,nec: -2,9917 cm²/m

Nº ramos: 2

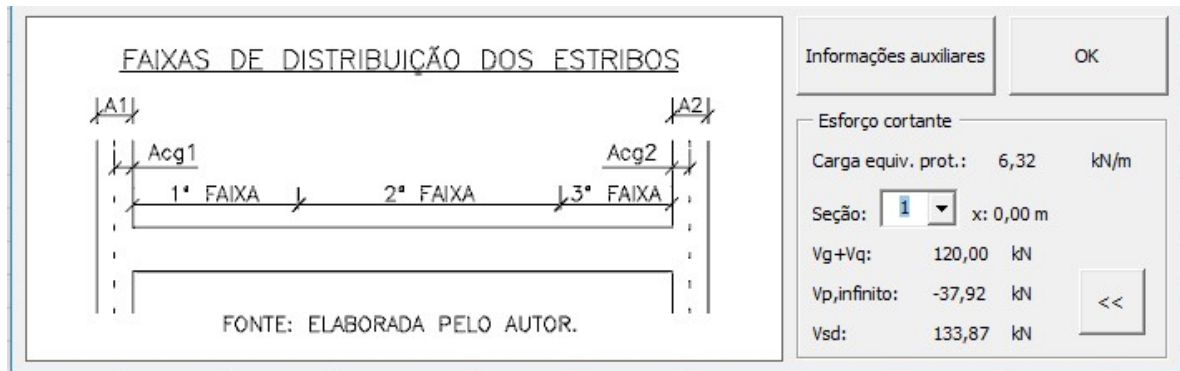
Bitola: 5 mm

Alojamento: 60 ϕ 5 c/ 11 (2 R)

Asw,adot: 3,62 cm²/m

Fonte: Elaborada pela autora.

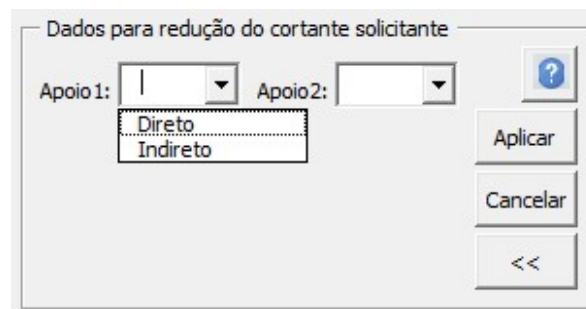
Figura 3-17 – Distribuição dos estribos em faixas



Fonte: Elaborada pela autora.

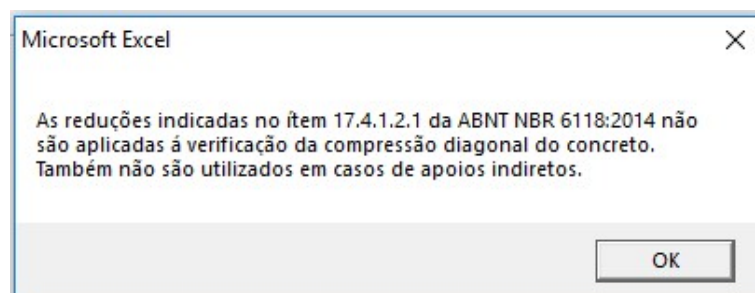
O programa oferece as opções de dimensionamento ao esforço cortante seguindo o modelo de cálculo I ou II, no segundo caso, deve-se informar a inclinação da diagonal comprimida de concreto. Há, também, a opção de redução do esforço cortante solicitante para cargas próximas aos apoios, no caso de apoio direto, para o cálculo da armadura transversal. Essas reduções não se aplicam à verificação da biela comprimida e nem a apoios indiretos. Ver Figura 3-18 e Figura 3-19.

Figura 3-18 – Dados para redução do cortante solicitante



Fonte: Elaborada pela autora.

Figura 3-19 – Janela de informação para o usuário



Fonte: Elaborada pela autora.

O usuário deve informar o diâmetro da bainha da armadura ativa para o cálculo da largura efetiva da seção. O programa verifica as diagonais comprimidas de concreto e calcula as armaduras transversais necessárias nas três faixas de distribuição. Devem-se informar os diâmetros desejados para os estribos, e a ferramenta, realiza o alojamento. É gerado um relatório com dos dados da seção, verificação da diagonal de concreto comprimida, força cortante solicitante máxima de cálculo de cada faixa, armadura transversal mínima, necessária e adotada, além do seu alojamento para cada faixa de distribuição dos estribos.

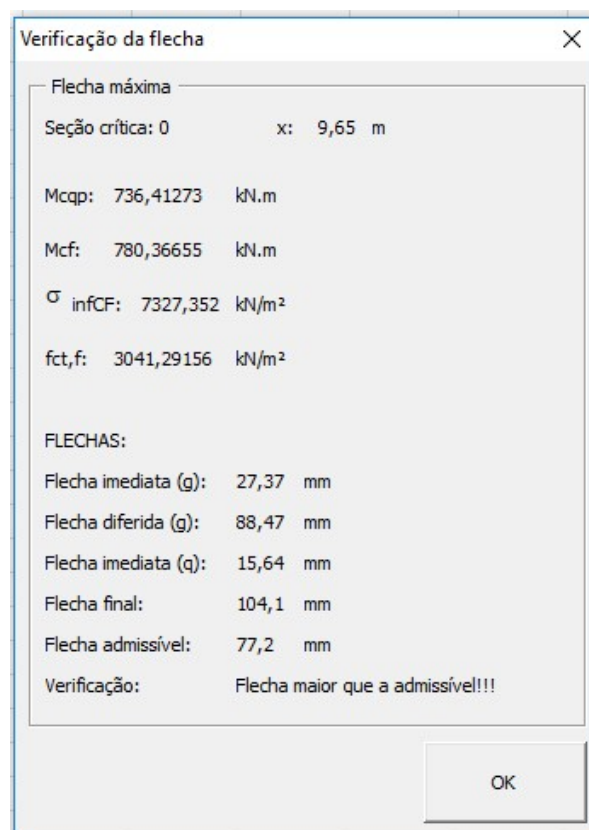
Finalmente, o usuário deve acessar a janela “Verificação da flecha”, pelo nono ícone da barra de ferramentas. Ver Figura 3-20 e Figura 3-21.

Figura 3-20 – Nono ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Figura 3-21 – Nono ícone da barra de ferramentas



Fonte: Elaborada pela autora.

Para a seção mais solicitada, o programa calcula as flechas imediatas para as cargas permanentes e acidentais, na combinação quase permanente. Então, calcula a flecha diferida, devida ao carregamento permanente. Finalmente, somando-se a flecha diferida com a flecha imediata acidental é obtida a flecha total e comparada à flecha admissível. É gerado um relatório com os valores das flechas imediata, diferida, total e admissível, indicando se foi atendido o estado-limite de deformação excessiva.

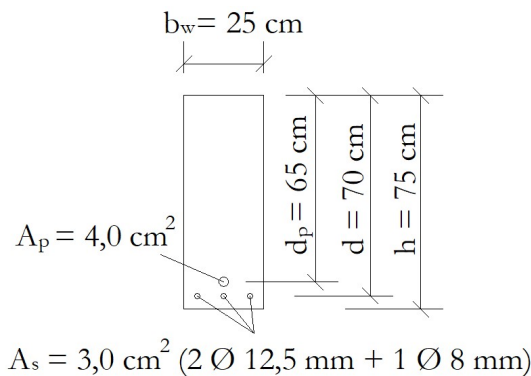
4 APLICAÇÕES NUMÉRICAS

Neste capítulo, são apresentadas aplicações numéricas presentes na literatura e desenvolvimento de cálculos, cujos resultados foram comparados com os obtidos no programa desenvolvido neste trabalho.

4.1 Aplicação numérica 1

França (2001) apresenta o cálculo da tensão na armadura passiva e abertura de fissuras para uma viga de seção transversal retangular, conforme Figura 4-1. Seja $\varepsilon_{pré} = 5 ‰$, $E_p = 200$ GPa, armadura aderente em bainha CP 190 RB, $E_s = 210$ GPa, $E_{cs} = 25$ GPa, $f_{ct} = f_{ctm} = 2,0$ MPa, e o momento fletor na combinação frequente igual a 245 kN.m.

Figura 4-1 – Seção transversal da viga



Fonte: Adaptada de França (2001).

4.1.1 Protensão parcial: verificação do estado-limite de abertura das fissuras

A equação (4.1) mostra que, para a combinação frequente, a tensão de tração na borda inferior da seção ultrapassa o $f_{ct} = 0,2 \text{ kN/cm}^2$ dado por França (2001), evidenciando a protensão parcial, e indicando a análise no estágio II.

$$\begin{aligned}
 \sigma = \frac{M_{CF}}{W_{inf}} + \frac{\varepsilon_{pré} E_p A_p}{A} + \frac{\varepsilon_{pré} E_p A_p \cdot e}{W_{inf}} &= \frac{24500}{23438} - \frac{0,005 \times 20000 \times 4}{1875} - \\
 &- \frac{0,005 \times 20000 \times 4 \times 27,5}{23438} = 0,36 \text{ kN/cm}^2
 \end{aligned} \quad (4.1)$$

Para desenvolvimento desta aplicação numérica no programa, foi adotado vão de 15 metros com carregamento permanente distribuído de 4,02 kN/m, que somado ao peso próprio da viga, resulta em $M_{CF} = 245$ kN.m. Para a obtenção da área de armadura ativa indicada na aplicação numérica, foi adotado o balanceamento de 40% do momento devido ao carregamento permanente, resultando no alojamento de 4 cordoalhas CP 190 RB 12,7 mm. Arbitrou-se $\varepsilon_{pré} = 5$ ‰, conforme indicado no problema.

Os resultados obtidos por França (2001) e pelo programa são apresentados na Tabela 4.1, observa-se que são muito próximos. A abertura característica das fissuras é inferior a 0,2 mm, portanto, respeita-se o estado-limite de abertura das fissuras.

Tabela 4.1 – Protensão parcial: aplicação numérica 1

	França (2001)	Programa
Área de armadura ativa A_p (cm ²)	4,0	4,056
Área de armadura passiva adotada A_s (cm ²)	3,0	3,14
Alojamento da armadura passiva	2 ϕ 12,5 mm + 1 ϕ 8 mm	4 ϕ 10 mm
Posição da linha neutra x (cm)	34,73	35,17
Deformação no concreto ε_c	0,42‰	0,42‰
Acréscimo de alongamento na armadura ativa $\Delta\varepsilon_p$	0,37‰	0,35‰
Alongamento na armadura passiva ε_s	0,43‰	0,42‰
Tensão no concreto σ_c (kN/cm ²)	1,05	1,05
Tensão na armadura ativa σ_p (kN/cm ²)	107,33	106,99
Tensão na armadura passiva σ_s (kN/cm ²)	8,97	8,73
Resultante do concreto N_c (kN)	456,23	461,40
Resultante da armadura ativa N_p (kN)	429,32	433,99
Resultante da armadura passiva N_s (kN)	26,91	27,41
Área de envolvimento das armaduras A_{cri} (cm ²)	359,38	312,5
Abertura característica das fissuras w (mm)	0,03	0,019

Fonte: Elaborada pela autora.

Verifica-se que ocorre uma diferença entre os valores das áreas das regiões de envolvimento das armaduras, o que se deve ao fato de terem sido adotados alojamentos com barras de

diâmetros distintos. França (2001) adota duas barras com diâmetro de 12,5 mm e uma de 8 mm, enquanto o programa utiliza 4 barras com diâmetro de 10 mm. Os cálculos das áreas de envolvimento das armaduras desenvolvido por França (2001) e pelo programa são apresentados em (4.2) e (4.3), respectivamente.

$$A_{cri} = b_w [(h - d) + 7,5\phi] = 25[(70 - 65) + 7,5 \times 1,25] = 359,38 \text{ cm}^2 \quad (4.2)$$

$$A_{cri} = b_w [(h - d) + 7,5\phi] = 25[(70 - 65) + 7,5 \times 1,0] = 312,5 \text{ cm}^2 \quad (4.3)$$

Salienta-se, também, que os valores característicos da abertura das fissuras foram calculados por equações distintas. França (2001) utiliza as expressões (4.4) e (4.5), da NBR 6118:1978, enquanto o programa segue a NBR 6118:2014, ver (4.6) e (4.7). Fator que influencia diretamente nos resultados.

$$w_k = \frac{1}{10} \frac{\phi_i}{(2\eta_b - 0,75)} \frac{\sigma_{si}}{E_{si}} \frac{3\sigma_{si}}{f_{ctm}} = \frac{1}{10} \times \frac{12,5\text{mm}}{(2 \times 1,5 - 0,75)} \times \frac{8,97}{21000} \times \frac{3 \times 8,97}{0,2} = 0,03\text{mm} \quad (4.4)$$

$$w_k = \frac{1}{10} \frac{\phi_i}{(2\eta_b - 0,75)} \frac{\sigma_{si}}{E_{si}} \left(\frac{4}{\rho_{ri}} + 45 \right) =$$

$$= \frac{1}{10} \times \frac{12,5\text{mm}}{(2 \times 1,5 - 0,75)} \times \frac{8,97}{21000} \times \left(\frac{4}{(3/358,38)} + 45 \right) = 0,12\text{mm} \quad (4.5)$$

$$w_k = \frac{\phi_i}{12,5\eta_1} \frac{\sigma_{si}}{E_{si}} \frac{3\sigma_{si}}{f_{ctm}} = \frac{10\text{mm}}{12,5 \times 2,25} \times \frac{8,73}{21000} \times \frac{3 \times 8,73}{0,2} = 0,019\text{mm} \quad (4.6)$$

$$w_k = \frac{\phi_i}{12,5\eta_1} \frac{\sigma_{si}}{E_{si}} \left(\frac{4}{\rho_{ri}} + 45 \right) = \frac{10\text{mm}}{12,5 \times 2,25} \times \frac{8,73}{21000} \times \left(\frac{4}{(3,14/312,5)} + 45 \right) = 0,065\text{mm} \quad (4.7)$$

4.1.2 Verificação do estado-limite de deformação excessiva

Inicialmente, o programa calcula o carregamento equivalente à protensão, apresentado em (4.8)

$$g_{eq} = \frac{8(\varepsilon_{pré} E_p A_p) e}{L^2} = \frac{8 \times (0,005 \times 20000 \times 4,056) \times 0,27}{15^2} = 3,89 \text{ kN/m} \quad (4.8)$$

Então, pela equação da linha elástica, o programa calcula o deslocamento na seção mais solicitada da viga, considerando o peso próprio g_{pp} , carregamento permanente g_1 e o alívio dos cabos g_{eq} , ver (4.9) e (4.10). Neste exemplo, particularmente, não há carregamento acidental. Caso houvesse, seria calculada também a flecha por ele provocada, considerando a combinação quase permanente.

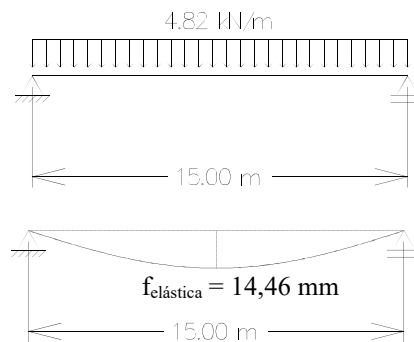
$$g_{total} = g_{pp} + g_1 - g_{eq} = 4,69 + 4,02 - 3,89 = 4,82 \text{ kN/m} \quad (4.9)$$

$$f_{elástica,g} = 14,46 \text{ mm} \quad (4.10)$$

A flecha elástica calculada pelo programa foi comparada com a flecha analítica (4.11), considerando a seção bruta de concreto, ver Figura 4-2. Como se observa, a flecha obtida é a mesma.

$$f_{elástica,g} = \frac{5 \times g_{total} \times L^4}{384 E_{cs} I_c} = \frac{5 \times 4,82 \times 15^4}{384 \times 25 \times 10^6 \times 0,008789} = 0,01446 \text{ m} \quad (4.11)$$

Figura 4-2 – Deslocamento



Fonte: Elaborada pela autora.

Como se trata de protensão parcial e, portanto, seção fissurada, deve-se calcular a rigidez equivalente para obtenção das flechas imediatas, conforme a equação (4.12). Para tanto, calcula-se o momento de fissuração pelas expressões (4.13) e (4.14). Para o cálculo do momento de inércia no estágio II, utiliza-se a equação

(4.15). Considera-se a combinação quase permanente para a determinação da posição da linha neutra, $x = 35,17 \text{ cm}$.

$$\begin{aligned} (EI)_{eq} &= E_{cs} \left\{ \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 I_c + \left[1 - \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 \right] I_{II} \right\} = \\ &= 2500 \times \left\{ \begin{aligned} &\left(\frac{20709 - 16021}{24500 - 16021} \right)^3 \times 878906,3 + \\ &+ \left[1 - \left(\frac{20709 - 16021}{24500 - 16021} \right)^3 \right] \times 422533,1 \end{aligned} \right\} = \\ &= 2500 \times 499618,6 = 1,249 \times 10^9 \text{ kN.cm}^2 \end{aligned} \quad (4.12)$$

$$M_0 = P_{inf} \left(e + \frac{W_c}{A_c} \right) = 405,6 \left(0,27 + \frac{0,023438}{0,1875} \right) = 160,21 \text{ kNm} = 16021 \text{ kNcm} \quad (4.13)$$

$$M_r = M_0 + W_c f_{ct} = 160,21 + 0,023438 \times 2000 = 207,09 \text{ kNm} = 20709 \text{ kNcm} \quad (4.14)$$

$$I_{II} = \frac{b_c x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \frac{E_p}{E_{cs}} A_p (d_p - x)^2 = \quad (4.15)$$

$$= \frac{25 \times 35,1736^3}{3} + \frac{21000}{2500} \times 3,14 \times (70 - 35,1736)^2 + \frac{20000}{2500} \times 4,056 \times \\ \times (64,5 - 35,1736)^2 = 422533,1 \text{ cm}^4$$

Então, a flecha imediata, na seção mais solicitada, pode ser calculada por (4.16).

$$f_{\text{imediata},g} = f_{\text{elástica},g} \frac{E_{cs}I_c}{(EI)_{eq}} = 14,46 \text{ mm} \times \frac{2500 \times 878906,3}{2500 \times 499618,6} = 25,44 \text{ mm} \quad (4.16)$$

A flecha diferida, referente ao carregamento permanente, é dada por (4.17).

$$f_{\text{diferida},g} = (1 + \varphi_{\infty}) f_{\text{imediata},g} = (1 + 2,2582) \times 25,44 = 82,90 \text{ mm} \quad (4.17)$$

Como neste exemplo, não há carregamento acidental, a flecha total é a própria flecha diferida, o programa, então, compara o valor calculado com o limite estabelecido pela NBR 6118:2014, considerando a aceitabilidade sensorial visual. Ver (4.18) e (4.19).

$$f_{\text{total}} = f_{\text{diferida},g} = 82,90 \text{ mm} \quad (4.18)$$

$$f_{\text{admissível}} = L/250 = 15000/250 = 60 \text{ mm} \quad (4.19)$$

Portanto, considerando a aceitabilidade sensorial visual, a flecha total é superior à admissível, logo, não está atendido o estado-limite de deformação excessiva. A Tabela 4.2 apresenta os resultados obtidos pelo programa na verificação.

Tabela 4.2 – Verificação da flecha: aplicação numérica 1

	Programa
Momento de fissuração (kN.cm)	20.709
Posição da linha neutra x_{II} (cm)	35,17
Rigidez da seção no estágio I (kN.cm ²)	2,197 x 10 ⁹
Rigidez da seção no estágio II (kN.cm ²)	1,056 x 10 ⁹
Rigidez equivalente (EI) _{eq} (kN.cm ²)	1,249 x 10 ⁹
Flecha imediata (mm)	25,44
Flecha diferida (mm)	82,90
Flecha total (mm)	82,90
Flecha admissível (mm)	60

Fonte: Elaborada pela autora.

Uma possível solução para redução da flecha, de modo que esta se torne inferior à admissível, é o acréscimo de mais uma cordoalha, totalizando cinco cordoalhas CP 190 RB 12,7 mm. Neste caso, seriam obtidos os resultados expostos na Tabela 4.3.

Tabela 4.3 – Verificação da flecha com 5 cordoalhas CP 190 RB 12,7

	Programa
Flecha imediata (mm)	11,52
Flecha diferida (mm)	37,55
Flecha total (mm)	37,55
Flecha admissível (mm)	60

Fonte: Elaborada pela autora.

4.1.3 Dimensionamento à força cortante

No dimensionamento à força cortante, a distribuição dos estribos é realizada em três faixas, a primeira até a coordenada $x_{f1} = x_{i2} = 5$ m, a segunda até $x_{f2} = x_{i3} = 10,0$ m e a terceira até o segundo apoio.

Para consideração do efeito da armadura ativa, utiliza-se a carga equivalente ao esforço de protensão, apresentada em (4.8). Então, são obtidas as forças cortantes solicitantes máximas de cálculo para cada faixa, conforme as equações (4.20) e (4.21).

$$V_{sd,1} = V_{sd,3} = 1,4 \times \frac{(g_{pp} + g_1) L}{2} - 0,9 \times \frac{g_{eq} L}{2} = 65,21 \text{ kN} \quad (4.20)$$

$$V_{sd,2} = 65,21 - 1,4 \times (g_{pp} + g_1) \times 5 + 0,9 \times g_{eq} \times 5 = 21,74 \text{ kN} \quad (4.21)$$

A força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas é dada por (4.22), e observa-se que é maior que a força cortante solicitante máxima de cálculo.

$$V_{Rd2} = 0,27 \times \left(1 - \frac{f_{ck}}{250}\right) f_{cd} b_w d = 0,27 \times \left(1 - \frac{17,2}{250}\right) \times \frac{1,72}{1,4} \times 25 \times 70 = 540,56 \text{ kN} \quad (4.22)$$

O momento fletor máximo de cálculo é $M_{sd,max} = 343$ kN.m e o momento fletor que anula σ_p na borda tracionada por M_{sd} , M_0 , é dado por (4.23).

$$M_0 = 0,9 \cdot P_{inf} \left(e + \frac{W_c}{A_c}\right) = 0,9 \times 405,6 \left(0,27 + \frac{0,023438}{0,1875}\right) = 144,19 \text{ kNm} \quad (4.23)$$

A parcela de força cortante absorvida por mecanismos complementares ao da treliça V_c é dada por (4.24).

$$V_c = 0,009 f_{ck}^{2/3} b_w d \left(1 + \frac{M_0}{M_{sd,max}} \right) = \quad (4.24)$$

$$= 0,009 \times 17,2^{2/3} \times 25 \times 70 \times \left(1 + \frac{144,19}{343} \right) = 149,06 \text{ kN}$$

Então, é calculada a área de armadura transversal necessária para a primeira e terceira faixas, ver (4.25).

$$A_{sw1/m} = A_{sw3/m} = \frac{V_{sd} - V_c}{0,9 d f_{ywd}} = \frac{65,21 - 149,07}{0,9 \times 0,7 \times 43,48} = -3,06 \text{ cm}^2/m \quad (4.25)$$

Observa-se que V_{sd} é menor que V_c , e, portanto, o valor da área de aço necessária é negativo. Logo, é necessário adotar, nas três faixas, a armadura mínima dada por (4.26), em que α é a inclinação dos estribos em relação ao eixo longitudinal e $s = 100$ cm. Portanto, as armaduras transversais podem ser alojadas em 2 ramos de $\phi 5$ mm c/19.

$$A_{swmin} = 0,2 \frac{f_{ct,m}}{f_{ywk}} b_w . s . \text{sen } \alpha = 0,2 \times \frac{2,0}{500} \times 25 \times 100 \times \text{sen } 90 = 2,0 \text{ cm}^2/m \quad (4.26)$$

Tabela 4.4 apresenta os resultados obtidos pelo programa no dimensionamento à força cortante.

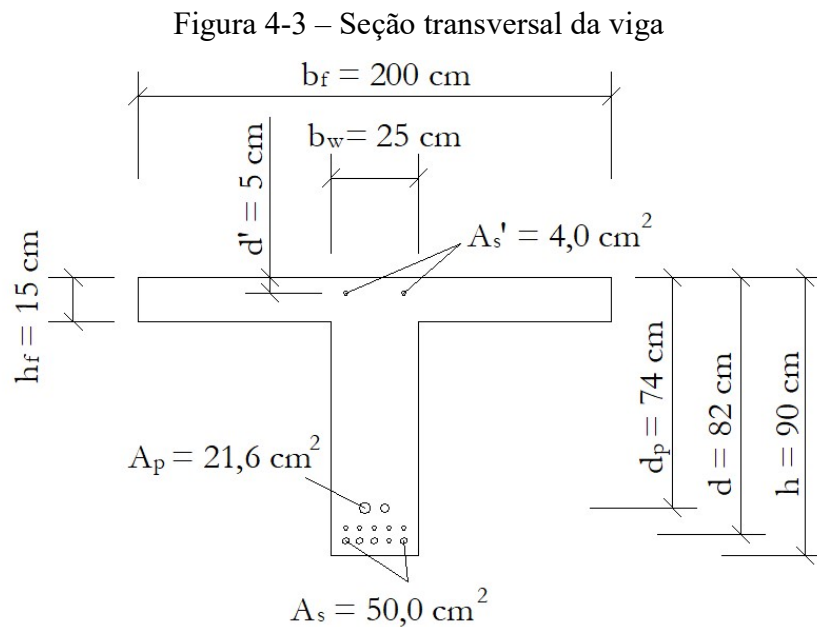
Tabela 4.4 – Dimensionamento à força cortante: aplicação numérica 1

	Programa
Carga equivalente ao esforço de protensão (kN/m)	3,89
Força cortante máxima solicitante de cálculo para 1ª e 3ª faixa (kN)	65,21
Força cortante máxima solicitante de cálculo para 2ª faixa (kN)	21,74
Força cortante resistente de cálculo relativa à ruína das diagonais comprimidas (kN)	540,56
Parcela de força cortante absorvida por mecanismos complementares ao da treliça (kN)	149,06
Área de armadura transversal necessária para 1ª e 3ª faixa (cm ² /m)	-3,06
Área de armadura transversal mínima (cm ² /m)	2,0

Fonte: Elaborada pela autora.

4.2 Aplicação numérica 2

França (2001) apresenta, também, o cálculo da tensão na armadura passiva e abertura de fissuras para uma viga de seção T, conforme Figura 4-3. Seja o vão da viga $L = 18$ m, a força de protensão após as perdas $P_{inf} = 2238$ kN, $E_p = 200$ GPa, armadura não aderente, $E_s = 210$ GPa, $E_{cs} = 25$ GPa, $f_{ctm} = 2,4$ MPa, $f_{ct} = 1,2 f_{ctm} = 2,9$ MPa, $g_{0k} = 23,5$ kN/m, $g_{1k} = 14,0$ kN/m, $q_k = 28$ kN/m e $\psi_1 = \psi_2 = 0,3$.



Fonte: Adaptada de França (2001).

4.2.1 Protensão parcial: verificação do estado-limite de abertura das fissuras

Para a obtenção da área de armadura ativa indicada na aplicação numérica, foi adotado o balanceamento de 77% do momento devido ao carregamento permanente, resultando no alojamento de 15 cordoalhas CP 190 RB 15,2 mm. Foi adotada a mesma armadura passiva indicada no problema.

A equação (4.27) mostra que, para a combinação frequente, a tensão de tração na borda inferior da seção ultrapassa o $f_{ct} = 0,29$ kN/cm² dado por França (2001), evidenciando a protensão parcial, e indicando a análise no estágio II.

$$\sigma = \frac{M_{CF}}{W_{inf}} + \frac{P_{inf}}{A} + \frac{P_{inf}e}{W_{inf}} = \frac{185897}{50185} - \frac{2238}{4875} - \frac{2238 \times 49}{50185} = 1,06 \text{ kN/cm}^2 \quad (4.27)$$

Os resultados obtidos por França (2001) e pelo programa são apresentados na Tabela 4.5, observa-se que os resultados são muito próximos. Salienta-se que, como no exemplo do item 4.1, as armaduras passivas foram alojadas em diâmetros diferentes e os valores característicos das aberturas das fissuras foram calculados por equações distintas. A abertura característica das fissuras é inferior a 0,2 mm, portanto, está respeitado o estado-limite de abertura das fissuras.

Tabela 4.5 – Protensão parcial: aplicação numérica 2

	França (2001)	Programa
Área de armadura ativa A_p (cm ²)	21,6	21,53
Área de armadura passiva adotada A_s (cm ²)	50,0	50,27
Área de armadura dupla adotada A_s' (cm ²)	4,0	4,0
Posição da linha neutra x (cm)	33,83	33,66
Deformação no concreto ε_c	0,47‰	0,475‰
Tensão na armadura passiva σ_s (kN/cm ²)	14,12	14,34
Tensão na armadura dupla σ_s' (kN/cm ²)	-8,45	-8,49
Abertura característica das fissuras w (mm)	0,06	0,048

Fonte: Elaborada pela autora.

4.2.2 Verificação do estado-limite de deformação excessiva

Inicialmente, o programa calcula o carregamento equivalente à protensão, apresentado em (4.28)

$$g_{eq} = \frac{8 \times P_{inf} \times e}{L^2} = \frac{8 \times 2238 \times 0,49}{18^2} = 27,07 \text{ kN/m} \quad (4.28)$$

Então, pela equação da linha elástica, o programa calcula o deslocamento, na seção mais solicitada da viga, considerando o peso próprio, o carregamento permanente e o alívio dos cabos g_{eq} , ver (4.29) e (4.30). Calcula, também, a flecha devida ao carregamento acidental q , ver (4.31) e (4.32). Considera-se a combinação quase permanente.

$$g_{total} = g_0 + g_1 - g_{eq} = 23,5 + 14 - 27,07 = 10,43 \text{ kN/m} \quad (4.29)$$

$$f_{elástica,g} = 17,43 \text{ mm} \quad (4.30)$$

$$q_d = \Psi_2 \times q_d = 0,3 \times 28 = 8,4 \text{ kN/m} \quad (4.31)$$

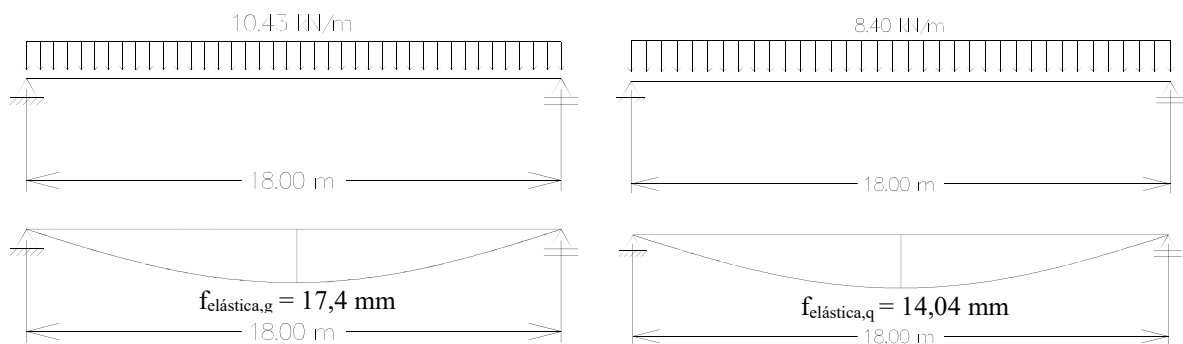
$$f_{elástica,q} = 14,04 \text{ mm} \quad (4.32)$$

As flechas elásticas calculadas pelo programa foram comparadas com as flechas analíticas, (4.33) e (4.34), considerando a seção bruta de concreto, ver Figura 4-4. Como se observa, as flechas obtidas são as mesmas.

$$f_{elástica,g} = \frac{5 \times g_{total} \times L^4}{384 E_{cs} I_c} = \frac{5 \times 10,43 \times 18^4}{384 \times 25 \times 10^6 \times 0,032717} = 0,01743 \text{ m} \quad (4.33)$$

$$f_{elástica,q} = \frac{5 \times q_d \times L^4}{384 E_{cs} I_c} = \frac{5 \times 8,4 \times 18^4}{384 \times 25 \times 10^6 \times 0,032717} = 0,01404 \text{ m} \quad (4.34)$$

Figura 4-4 – Deslocamentos



Fonte: Elaborada pelo autor.

Como se trata de protensão parcial e, portanto, seção fissurada, deve-se calcular a rigidez equivalente para obtenção das flechas imediatas, conforme a equação (4.35). Para tanto, calcula-se o momento de fissuração pelas expressões (4.36) e (4.37). Para o cálculo do momento de inércia no estágio II, utiliza-se a equação (4.38). Considera-se a combinação quase permanente para a determinação da posição da linha neutra, particularmente neste exemplo, foi utilizado mesmo valor para ψ_1 e ψ_2 , e portanto $x = 33,66$ cm.

$$\begin{aligned} (EI)_{eq} &= E_{cs} \left\{ \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 I_c + \left[1 - \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 \right] I_{II} \right\} = \\ &= 2500 \times \left\{ \left(\frac{142779 - 132670}{185897 - 132670} \right)^3 \times 3271695 + \right. \\ &\quad \left. + \left[1 - \left(\frac{142779 - 132670}{185897 - 132670} \right)^3 \right] \times 3173960 \right\} = \\ &= 2500 \times 3174629 = 7,937 \times 10^9 \text{ kN.cm}^2 \end{aligned} \quad (4.35)$$

$$M_0 = P_{inf} \left(e + \frac{W_c}{A_c} \right) = 2237,5 \left(0,49 + \frac{0,050185}{0,4875} \right) = 1326,7 \text{ kNm} = 132670 \text{ kNcm} \quad (4.36)$$

$$M_r = M_0 + W_c f_{ct} = 1326,7 + 0,050185 \times 2900 = 1427,79 \text{ kNm} = 142779 \text{ kNcm} \quad (4.37)$$

$$\begin{aligned}
I_{II} &= \frac{(b_f - b_w)h_f^3}{12} + (b_f - b_w)h_f \left(x - \frac{h_f}{2}\right)^2 + \frac{b_w x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \\
&+ \left(\frac{E_s}{E_{cs}} - 1\right) A_s' (x - d')^2 = \frac{(200 - 25) \times 15^3}{12} + (200 - 25) \times 15 \times \\
&\times \left(33,6559 - 15/2\right)^2 + \frac{25 \times 33,6559^3}{3} + \frac{21000}{2500} \times 50,27 \times (82 - 33,6559)^2 + \\
&+ \left(\frac{21000}{2500} - 1\right) \times 4 \times (33,6559 - 5)^2 = 3173960 \text{ cm}^4
\end{aligned} \tag{4.38}$$

Então, as flechas imediatas, na seção mais solicitada, podem ser calculadas por (4.39) e (4.40).

$$f_{imediate,g} = f_{elástica,g} \frac{E_{cs} I_c}{(EI)_{eq}} = 17,43 \text{ mm} \times \frac{2500 \times 3271695}{2500 \times 3174629} = 17,96 \text{ mm} \tag{4.39}$$

$$f_{imediate,q} = f_{elástica,q} \frac{E_{cs} I_c}{(EI)_{eq}} = 14,04 \text{ mm} \times \frac{2500 \times 3271695}{2500 \times 3174629} = 14,47 \text{ mm} \tag{4.40}$$

A flecha diferida, referente ao carregamento permanente, é dada por (4.41).

$$f_{diferida,g} = (1 + \varphi_{\infty}) f_{imediate,g} = (1 + 2,2715) \times 17,96 = 58,77 \text{ mm} \tag{4.41}$$

A flecha total é a soma da flecha diferida, devida ao carregamento permanente, com a flecha imediata devida à carga acidental, o programa, então, compara o valor calculado com o limite estabelecido pela NBR 6118:2014, considerando a aceitabilidade sensorial visual. Ver (4.42) e (4.43).

$$f_{total} = f_{diferida,g} + f_{imediate,q} = 58,77 + 14,47 = 73,24 \text{ mm} \tag{4.42}$$

$$f_{admissível} = L/250 = 18000/250 = 72 \text{ mm} \tag{4.43}$$

Portanto, considerando a aceitabilidade sensorial visual, a flecha total é ligeiramente superior à admissível. A Tabela 4.6 apresenta os resultados obtidos pelo programa na verificação.

Tabela 4.6 – Verificação da flecha: aplicação numérica 2

	Programa
Momento de fissuração (kN.cm)	142.779
Posição da linha neutra x_{II} (cm)	33,66
Rigidez da seção no estágio I (kN.cm ²)	8,179 x 10 ⁹
Rigidez da seção no estágio II (kN.cm ²)	7,935 x 10 ⁹
Rigidez equivalente (EI) _{eq} (kN.cm ²)	7,937 x 10 ⁹
Flecha imediata devida ao carregamento permanente (mm)	17,96
Flecha imediata devida ao carregamento acidental (mm)	14,47
Flecha diferida (mm)	58,77
Flecha total (mm)	73,24
Flecha admissível (mm)	72

Fonte: Elaborada pela autora.

4.2.3 Dimensionamento à força cortante

No dimensionamento à força cortante, a distribuição dos estribos é realizada em três faixas, a primeira até a coordenada $x_{f1} = x_{f2} = 6$ m, a segunda até $x_{f2} = x_{f3} = 12,0$ m e a terceira até o segundo apoio.

Para consideração do efeito da armadura ativa, utiliza-se a carga equivalente ao esforço de protensão, apresentada em (4.28). Então, são obtidas as forças cortantes solicitantes máximas de cálculo para cada faixa, conforme as equações (4.44) e (4.45).

$$V_{sd,1} = V_{sd,3} = 1,4 \times \frac{(g_0 + g_1 + q) L}{2} - 0,9 \times \frac{g_{eq} L}{2} = 606,04 \text{ kN} \quad (4.44)$$

$$V_{sd,2} = 606,04 - 1,4 \times (g_0 + g_1 + q) \times 6 + 0,9 \times g_{eq} \times 6 = 202,01 \text{ kN} \quad (4.45)$$

A força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas é dada por (4.46), e observa-se que é maior que a força cortante solicitante máxima de cálculo.

$$V_{Rd2} = 0,27 \times \left(1 - \frac{f_{ck}}{250}\right) f_{cd} b_w d = 0,27 \times \left(1 - \frac{22,6}{250}\right) \times \frac{2,26}{1,4} \times 25 \times 82 = 812,73 \text{ kN} \quad (4.46)$$

O momento fletor máximo de cálculo é $M_{sd,max} = 3713,88$ kN.m e o momento fletor que anula σ_p na borda tracionada por M_{sd} , M_0 , é dado por (4.47).

$$M_0 = 0,9 \cdot P_{inf} \left(e + \frac{W_c}{A_c}\right) = 0,9 \times 2238 \left(0,49 + \frac{0,050185}{0,4875}\right) = 1194,03 \text{ kNm} \quad (4.47)$$

A parcela de força cortante absorvida por mecanismos complementares ao da treliça V_c é dada por (4.48).

$$\begin{aligned} V_c &= 0,009 f_{ck}^{2/3} b_w d \left(1 + \frac{M_0}{M_{sd,max}} \right) = \\ &= 0,009 \times 22,6^{2/3} \times 25 \times 82 \times \left(1 + \frac{1194,03}{3713,88} \right) = 194,896 \text{ kN} \end{aligned} \quad (4.48)$$

Então, calcula-se a área de armadura transversal necessária para a primeira e terceira faixa, ver (4.49). Assim, adota-se estribos de 2 ramos de ϕ 10 mm c/12 para essas faixas.

$$A_{sw1/m} = A_{sw3/m} = \frac{V_{sd} - V_c}{0,9 d f_{ywd}} = \frac{606,04 - 194,896}{0,9 \times 0,82 \times 43,48} = 12,81 \text{ cm}^2/m \quad (4.49)$$

Passa-se, então, ao cálculo da área de armadura transversal necessária para a segunda faixa, ver (4.50).

$$A_{sw2/m} = \frac{V_{sd} - V_c}{0,9 d f_{ywd}} = \frac{202,01 - 194,896}{0,9 \times 0,82 \times 43,48} = 0,22 \text{ cm}^2/m \quad (4.50)$$

Logo, é necessário adotar, na segunda faixa, a armadura mínima dada por (4.51), em que α é a inclinação dos estribos em relação ao eixo longitudinal e $s = 100$ cm. Portanto, as armaduras transversais podem ser alojadas em 2 ramos de ϕ 6,3 mm c/25.

$$A_{swmin} = 0,2 \frac{f_{ct,m}}{f_{ywk}} b_w . s . \text{sen } \alpha = 0,2 \times \frac{2,4}{500} \times 25 \times 100 \times \text{sen } 90 = 2,4 \text{ cm}^2/m \quad (4.51)$$

A Tabela 4.7 apresenta os resultados obtidos pelo programa no dimensionamento à força cortante.

Tabela 4.7 – Dimensionamento à força cortante: aplicação numérica 2

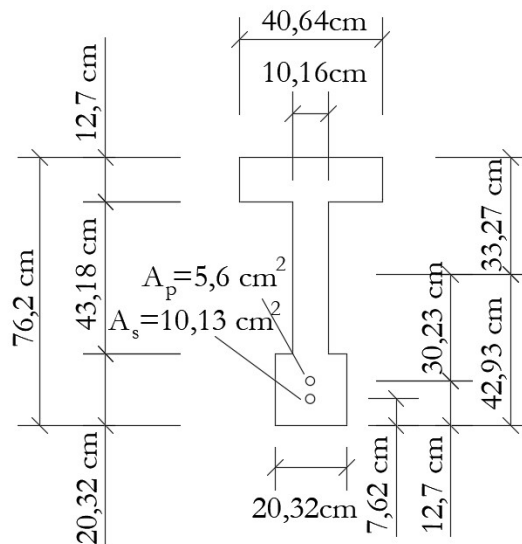
	Programa
Carga equivalente ao esforço de protensão (kN/m)	27,07
Força cortante máxima solicitante de cálculo para 1ª e 3ª faixa (kN)	606,04
Força cortante máxima solicitante de cálculo para 2ª faixa (kN)	202,01
Força cortante resistente de cálculo relativa à ruína das diagonais comprimidas (kN)	812,73
Parcela de força cortante absorvida por mecanismos complementares ao da treliça (kN)	194,89
Área de armadura transversal necessária para 1ª e 3ª faixa (cm ² /m)	12,81
Área de armadura transversal necessária para 2ª faixa (cm ² /m)	0,22
Área de armadura transversal mínima (cm ² /m)	2,4

Fonte: Elaborada pela autora.

4.3 Aplicação numérica 3

Nilson (1976) apresenta o cálculo da posição da linha neutra e tensões no concreto e nas armaduras passivas e ativas para a seção transversal de uma viga submetida à protensão parcial, conforme Figura 4-5. Trata-se de uma viga com protensão aderente, seja o momento devido a carga permanente igual a 52 kNm, o momento devido às cargas acidentais igual a 259 kNm, e o momento devido ao peso próprio igual a 113 kNm, a área de armadura ativa igual a 5,5 cm² e de armadura passiva igual a 10,1 cm², $E_p = 186$ GPa, $E_s = 200$ GPa e $E_{cs} = 24,9$ GPa.

Figura 4-5 – Seção transversal da viga



Fonte: Adaptada de Nilson (1976).

4.3.1 Protensão parcial: verificação do estado-limite de abertura das fissuras

Para a obtenção da área de armadura ativa indicada na aplicação numérica, foi adotado o balanceamento de 90% do momento devido ao carregamento permanente, resultando no alojamento de 4 cordoalhas CP 190 RB 15,2 mm. Foi adotada a mesma armadura passiva indicada no problema.

Os resultados obtidos por Nilson (1976) e pelo programa são apresentados na Tabela 4.8, observa-se que os resultados são relativamente próximos. O programa realiza o equilíbrio de forças e momentos na seção transversal no estágio II para a combinação frequente. Enquanto Nilson (1976) realiza o cálculo da posição da linha neutra utilizando o

equilíbrio de momentos na seção homogeneizada sem considerar a área da seção transversal, mas apenas as tensões. Deve-se considerar que houve pequenas diferenças nas alturas úteis adotadas pelo programa no alojamento e aproximações nas transformações de unidades.

Tabela 4.8 – Protensão parcial: aplicação numérica 3

	Nilson (1976)	Programa
Área de armadura ativa A_p (cm ²)	5,6	5,6
Área de armadura passiva adotada A_s (cm ²)	10,13	10,13
Altura útil da armadura passiva d (cm)	68,6	69,1
Altura útil da armadura ativa d_p (cm)	63,5	63,4
Posição da linha neutra x (cm)	34	32,6
Tensão no concreto σ_c (kN/cm ²)	-1,5	-1,7
Tensão na armadura passiva σ_s (kN/cm ²)	11,1	15,4
Tensão na armadura ativa σ_p (kN/cm ²)	114,8	117,8

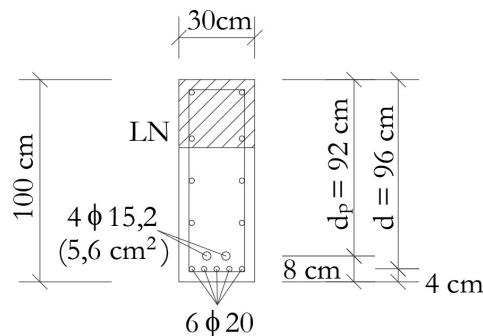
Fonte: Elaborada pela autora.

4.4 Aplicação numérica 4

Cholfe e Bonilha (2013) apresentam uma aplicação numérica do dimensionamento de uma viga com protensão parcial e verificação do estado-limite de abertura das fissuras, cujos resultados são comparados com os obtidos pelo programa. Trata-se de uma seção retangular dimensionada no estado-limite último resultando em $A_p = 5,6$ cm² (4 ϕ 15,2) e $A_s = 18,11$ cm² (6 ϕ 20; 18,90 cm²), ver Figura 4-6.

Utiliza-se o concreto C30 ($E_{cs} = 26071,59$ MPa), aço CP190 RB ($E_p = 200000$ MPa; $\epsilon_{pré} = 5,50$ ‰) e CA50 ($E_s = 210000$ MPa, $\alpha_e = E_s/E_c = 15$). As solicitações são $M_{g1k} = 350$ kN.m, $M_{g2k} = 227$ kN.m, $M_{q1k} = 220$ kN.m ($\psi_1 = 0,6$; $\psi_2 = 0,4$) e $M_{q2k} = 120$ kN.m ($\psi_1 = 0,7$; $\psi_2 = 0,6$).

Figura 4-6 – Seção transversal da viga



Fonte: Adaptada de Cholfe e Bonilha (2013).

Para o desenvolvimento deste exemplo no programa, são calculados o vão da viga e carregamentos que resultem nos momentos fletores apresentados por Cholfe e Bonilha (2013). Considerando $M_{g1k} = 350$ kN.m referente ao peso próprio da viga, calcula-se o seu vão, $L = 19,3$ m e obtém-se as cargas distribuídas $g_2 = 4,88$ kN/m, $q_1 = 4,72$ kN/m e $q_2 = 2,58$ kN/m. Para obtenção da área de armadura ativa indicada na aplicação numérica, foi adotado balanceamento de 40% do momento devido ao carregamento permanente.

A Tabela 4.9 apresenta, para o meio do vão, os resultados obtidos pelo programa referentes à força inicial e final de protensão, perdas, tensões nas bordas inferior e superior, áreas de armadura ativa e passiva, e, razão entre a posição da linha neutra e altura útil da armadura passiva no estado-limite último (x/d).

Tabela 4.9 – Resultados obtidos para o meio do vão

Viga submetida à protensão parcial	Programa
Força inicial de protensão (kN)	804,86
Perda por atrito (kN)	9,00
Perda por ancoragem (kN)	35,71
Perda por encurtamento (kN)	29,56
Perdas progressivas (kN)	24,91
Força final de protensão (kN)	705,68
Porcentagem total de perdas (%)	12,32
Tensão na borda inferior CF (kN/m ²)	7567,59
Tensão na borda superior CF (kN/m ²)	-12272,14
Armadura ativa estimada $A_{p,est}$ (cm ²)	4,89
Armadura passiva necessária A_s (cm ²)	17,245
ELU: x/d	0,368

Fonte: Elaborada pela autora.

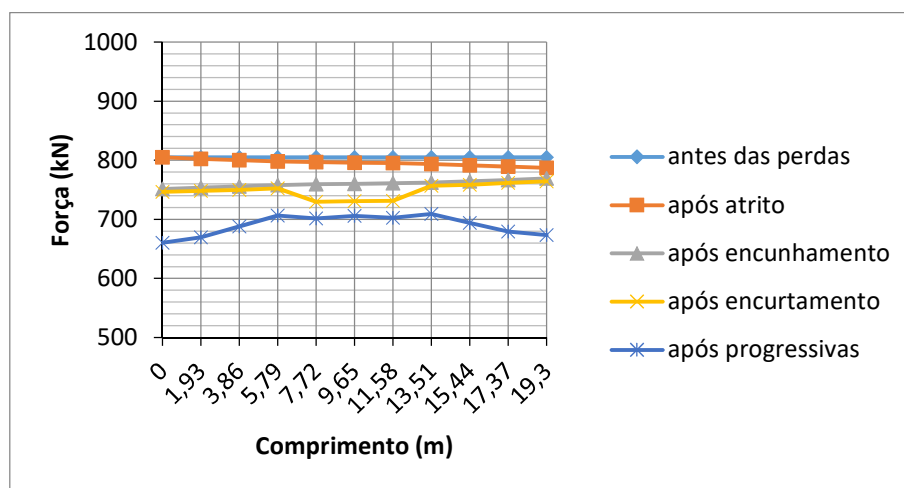
A Tabela 4.10 apresenta os valores obtidos pelo programa para a excentricidade do cabo, força inicial e final de protensão, perdas e tensões nas bordas inferior e superior nas seções de análise ao longo da viga. Enquanto a Figura 4-7 apresenta a força de protensão ao longo do cabo.

Tabela 4.10 – Perdas de protensão calculadas pelo programa

	Seção 1	Seção 2	Seção 3	Seção 4	Seção 5	Seção 6	Seção 7	Seção 8	Seção 9	Seção 10	Seção 11
Posição (m)	0	1,93	3,86	5,79	7,72	9,65	11,58	13,51	15,44	17,37	19,3
Excentricidade (m)	0	0,21	0,35	0,41	0,42	0,42	0,42	0,41	0,35	0,21	0
Força inicial de protensão (kN)	804,86	804,86	804,86	804,86	804,86	804,86	804,86	804,86	804,86	804,86	804,86
Perda por atrito (kN)	0	2,3	4,6	6,9	8,24	9,0	9,77	11,10	13,38	15,65	17,91
Perda por ancoragem (kN)	53,72	49,12	44,51	39,91	37,24	35,71	34,17	31,52	26,95	22,41	17,9
Perda por encurtamento (kN)	4,69	5,25	5,81	5,39	29,53	29,56	29,59	5,47	5,94	5,37	4,8
Perdas progressivas (kN)	85,99	78,47	62,20	46,24	28,32	24,91	28,77	47,51	64,76	82,1	90,63
Força final de protensão (kN)	660,47	669,73	687,74	706,42	701,53	705,68	702,56	709,26	693,83	679,33	673,63
% total de perdas	17,94	16,79	14,55	12,23	12,84	12,32	12,71	11,88	13,8	15,6	16,31
Tensão na borda inferior (kN/m ²)	-2201,6	627,89	3025,9	5094,1	6982,4	7567,6	6970,3	5061,1	2962,9	555,1	-2245,4
Tensão na borda superior (kN/m ²)	-2201,6	-5092,7	-7610,9	-9803,6	-11659	-12272	-11654	-9789,5	-7588,5	-5083,9	-2245,4

Fonte: Elaborada pela autora.

Figura 4-7 – Gráfico da força ao longo do cabo (programa)



Fonte: Elaborada pela autora.

A Tabela 4.11 apresenta, para as seções de análise no estado-limite último, os valores obtidos pelo programa para a força de protensão após as perdas, o momento solicitante de cálculo M_{sd} , o momento resistente de cálculo M_{Rd} , da razão entre a posição da linha neutra e a altura

útil da armadura passiva x/d , a área de armadura passiva necessária A_s e a mínima $A_{s,min}$. Não foi considerada a decalagem.

Tabela 4.11 – Estado-limite último (programa)

	Seção 1	Seção 2	Seção 3	Seção 4	Seção 5	Seção 6	Seção 7	Seção 8	Seção 9	Seção 10	Seção 11
Posição (m)	0	1,93	3,86	5,79	7,72	9,65	11,58	13,51	15,44	17,37	19,3
Força final de protensão (kN)	660,47	669,73	687,74	706,42	701,53	705,68	702,56	709,26	693,83	679,33	673,63
MSd = MRd (kN.m)	0	443,7	788,7	1035,2	1183,1	1232,4	1183,1	1035,2	788,7	443,7	0
x/d	0,008	0,124	0,224	0,302	0,351	0,368	0,351	0,302	0,224	0,124	0,008
Armadura passiva necessária A_s (cm ²)	-16,2	-5,2	3,8	10,8	15,7	17,3	15,7	10,7	3,7	-5,4	-16,4
Armadura passiva mínima $A_{s,min}$ (cm ²)	2,25	2,25	2,25	2,25	2,25	2,25	2,25	2,25	2,25	2,25	2,25

Fonte: Elaborada pela autora.

4.4.1 Protensão parcial: verificação do estado-limite de abertura das fissuras

A Tabela 4.12 apresenta os resultados da verificação da abertura das fissuras obtidos pelo programa e compara com Cholfe e Bonilha (2013).

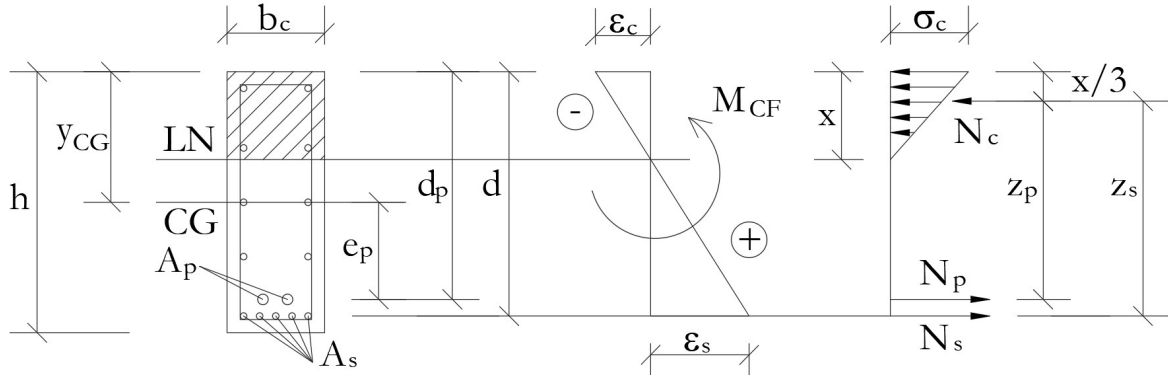
Tabela 4.12 – Protensão parcial: aplicação numérica 4

	Cholfe e Bonilha (2013)	Programa
Área de armadura ativa estimada A_p (cm ²)	5,6	4,89
Alojamento da armadura ativa	4 ϕ 15,2	4 ϕ 15,2
Área de armadura passiva necessária A_s (cm ²)	18,11	17,245
Alojamento da armadura passiva	6 ϕ 20	6 ϕ 20
Momento fletor na combinação frequente M_{CF} (kN.m)	781	780,4
Posição da linha neutra x (cm)	63	44,8
Tensão na armadura passiva σ_s (kN/cm ²)	6	13,7
Área de envolvimento das armaduras A_{cri} (cm ²)	570	570
Abertura característica das fissuras w (mm)	0,013	0,066

Fonte: Elaborada pela autora.

Observa-se grande diferença nos valores da posição da linha neutra e tensão na armadura, e consequentemente, no valor característico da abertura das fissuras. Tal fato se justifica pela diferença entre as equações de equilíbrio de momentos utilizadas. Ver Figura 4-8.

Figura 4-8 – Equilíbrio da seção



Fonte: Elaborada pela autora.

Do equilíbrio de forças, o programa utiliza as equações (4.52) a (4.54), para seções retangulares sem armadura dupla e com protensão não aderente. Sendo $\alpha_e = E_s/E_{cs}$.

$$N_c = N_p + N_s \quad (4.52)$$

$$\sigma_s \frac{b_c x}{2\alpha_e} \frac{x}{d-x} = \varepsilon_{pré} E_p A_p + A_s \cdot \sigma_s \quad (4.53)$$

$$\sigma_s = \frac{\varepsilon_{pré} E_p A_p}{\frac{b_c \cdot x}{2\alpha_e} \frac{x}{d-x} - A_s} \quad (4.54)$$

Do equilíbrio de momentos, o programa utiliza as equações (4.55) a (4.57), para seções retangulares sem armadura dupla e com protensão não aderente.

$$N_c(y_{cg} - x/3) + N_s(d - y_{cg}) + N_p(d_p - y_{cg}) = M_{CF} \quad (4.55)$$

$$\sigma_s \frac{b_c x}{2\alpha_e} \frac{x}{d-x} (y_{cg} - x/3) + A_s \sigma_s (d - y_{cg}) + \varepsilon_{pré} E_p A_p (d_p - y_{cg}) = M_{CF} \quad (4.56)$$

$$\sigma_s = \frac{M_{CF} - \varepsilon_{pré} E_p A_p (d_p - y_{cg})}{\frac{b \cdot x}{2\alpha_e} \frac{x}{d-x} (y_{cg} - x/3) + A_s (d - y_{cg})} \quad (4.57)$$

O programa tem como solução, o par x e σ_s que satisfaça simultaneamente às equações (4.54) e (4.57). No caso da aplicação numérica 4, $x = 44,78 \text{ cm}$ e $\sigma_s = 13,704 \text{ kN/cm}^2$.

Cholfé e Bonilha (2013) utilizam, do equilíbrio de forças, a expressão (4.58) que é semelhante a (4.54). Porém, utilizam, do equilíbrio de momentos, as expressões (4.59) a (4.64), em que

$$\sigma_s = \frac{A_p \varepsilon_{pré} E_p}{\frac{A_{cc}}{2} \cdot \frac{1}{\alpha_e} \cdot \frac{x}{d-x} - A_s} \quad (4.58)$$

$$N_s \times z_s + N_p \times z_p = \Delta M \quad (4.59)$$

$$\Delta M \approx M_{CF} - A_p \varepsilon_{pré} E_p \times e_p \quad (4.60)$$

$$z_s = d - \frac{x}{3} \quad (4.61)$$

$$z_p = d_p - \frac{x}{3} \quad (4.62)$$

$$A_s \sigma_s \times \left(d - \frac{x}{3}\right) + A_p \varepsilon_{pré} E_p \times \left(d_p - \frac{x}{3}\right) = \Delta M \quad (4.63)$$

$$\sigma_s = \frac{\Delta M - A_p \varepsilon_{pré} E_p \times \left(d_p - \frac{x}{3}\right)}{A_s \left(d - \frac{x}{3}\right)} \quad (4.64)$$

Observa-se que, na equação (4.64), a parcela relativa à protensão está sendo considerada duas vezes, tanto em N_p quanto em ΔM .

Outra diferença ocorre no valor de α_e . O programa adota α_e igual à razão entre o módulo de elasticidade do aço e o módulo de elasticidade secante do concreto. Já Cholfé e Bonilha (2013) utilizam α_e igual 15, valor permitido pela NBR 6118:2014 para o cálculo no estágio II. Ainda há o fato de que Cholfé e Bonilha (2013) arbitram o valor do pré-alongamento dos cabos como 5,5 %, enquanto o programa calcula este valor encontrando 6,4 %.

Substituindo ΔM por M_{CF} em (4.64), e utilizando $\alpha_e = E_s/E_{cs} = 8,0547$ e $\varepsilon_{pré} = 6,4$ ‰. O par que satisfaria simultaneamente (4.58) e (4.64), seria $x = 44,78$ cm e $\sigma_s = 13,7$ kN/cm², mesma solução encontrada pelo programa.

4.4.2 Verificação do estado-limite de deformação excessiva

Inicialmente, o programa calcula o carregamento equivalente à protensão, apresentado em (4.65)

$$g_{eq} = \frac{8 \times P_{inf} \times e}{L^2} = \frac{8 \times 705,68 \times 0,42}{19,32^2} = 6,37 \text{ kN/m} \quad (4.65)$$

Então, pela equação da linha elástica, o programa calcula o deslocamento, na seção mais solicitada da viga, considerando as cargas permanentes o peso próprio g_1 , o carregamento permanente g_2 e o alívio dos cabos g_{eq} , ver (4.66) e (4.67). Calcula, também, a flecha devida ao carregamento acidental, ver (4.68) e (4.69). Considera-se a combinação quase permanente.

$$g_{total} = g_1 + g_2 - g_{eq} = 7,5 + 4,88 - 6,37 = 6,01 \text{ kN/m} \quad (4.66)$$

$$f_{elástica,g} = 16,66 \text{ mm} \quad (4.67)$$

$$q_{total} = \Psi_2 \times q_1 + \Psi_2 \times q_2 = 0,4 \times 4,72 + 0,6 \times 2,58 = 3,44 \text{ kN/m} \quad (4.68)$$

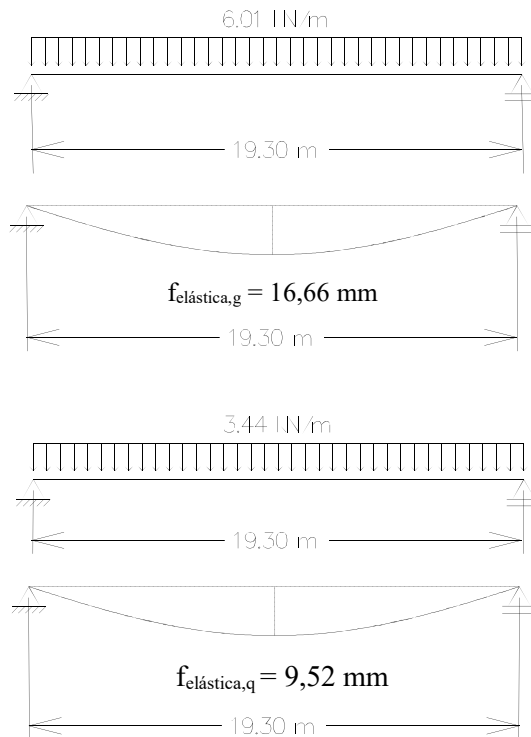
$$f_{elástica,q} = 9,52 \text{ mm} \quad (4.69)$$

As flechas elásticas calculadas pelo programa foram comparadas com as analíticas, (4.70) e (4.71), considerando a seção bruta de concreto, ver Figura 4-9. Como se observa, as flechas obtidas são as mesmas.

$$f_{elástica,g} = \frac{5 \times g_{total} \times L^4}{384 E_{cs} I_c} = \frac{5 \times 6,01 \times 19,3^4}{384 \times 26071590 \times 0,025} = 0,01666 \text{ m} \quad (4.70)$$

$$f_{elástica,q} = \frac{5 \times q_{total} \times L^4}{384 E_{cs} I_c} = \frac{5 \times 3,44 \times 19,3^4}{384 \times 26071590 \times 0,025} = 0,00952 \text{ m} \quad (4.71)$$

Figura 4-9 – Deslocamentos



Fonte: Elaborada pela autora.

Como se trata de protensão parcial e, portanto, seção fissurada, deve-se calcular a rigidez equivalente para obtenção das flechas imediatas, conforme a equação (4.72). Para tanto, calcula-se o momento de fissuração pelas expressões (4.73) e (4.74). Para o cálculo do momento de inércia no estágio II, utiliza-se a equação (4.75). Considera-se a combinação quase permanente para a determinação da posição da linha neutra, $x = 47,02$ cm.

$$(EI)_{eq} = E_{cs} \left\{ \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 I_c + \left[1 - \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 \right] I_{II} \right\} =$$

$$= 2607,16 \times \left\{ \left(\frac{56605 - 41399}{78036,66 - 41399} \right)^3 \times 2500000 + \left[1 - \left(\frac{56605 - 41399}{78036,66 - 41399} \right)^3 \right] \times 1406817 \right\} = \quad (4.72)$$

$$= 2607,16 \times 1521510 = 3,967 \times 10^9 \text{ kN.cm}^2$$

$$M_0 = P_{inf} \left(e + \frac{W_c}{A_c} \right) = 705,68 \left(0,42 + \frac{0,05}{0,3} \right) = 413,99 \text{ kNm} = 41399 \text{ kNcm} \quad (4.73)$$

$$M_r = M_0 + W_c f_{ct} = 413,99 + 0,05 \times 3041,29 = 566,05 \text{ kNm} = 56605 \text{ kNcm} \quad (4.74)$$

$$I_{II} = \frac{b_c x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 = \frac{30 \times 47,02^3}{3} + \frac{21000}{2607,16} \times 18,85 \times (96 - 47,02)^2 =$$

$$= 1406817 \text{ cm}^4 \quad (4.75)$$

Então, as flechas imediatas, na seção mais solicitada, podem ser calculadas por (4.76) e (4.77).

$$f_{imediate,g} = f_{elástica,g} \frac{E_{cs} I_c}{(EI)_{eq}} = 16,66 \text{ mm} \times \frac{2607,16 \times 2500000}{2607,16 \times 1521510} = 27,37 \text{ mm} \quad (4.76)$$

$$f_{imediate,q} = f_{elástica,q} \frac{E_{cs} I_c}{(EI)_{eq}} = 9,52 \text{ mm} \times \frac{2607,16 \times 2500000}{2607,16 \times 1521510} = 15,65 \text{ mm} \quad (4.77)$$

A flecha diferida, referente ao carregamento permanente, é dada por (4.78).

$$f_{diferida,g} = (1 + \varphi_\infty) f_{imediate,g} = (1 + 2,2322) \times 27,37 = 88,47 \text{ mm} \quad (4.78)$$

A flecha total é a da soma da flecha diferida, devida ao carregamento permanente, com a flecha imediata devida à carga acidental, o programa, então, compara o valor calculado com o limite estabelecido pela NBR 6118:2014, considerando a aceitabilidade sensorial visual. Ver (4.79) e (4.80).

$$f_{total} = f_{diferida,g} + f_{imediate,q} = 88,47 + 15,65 = 104,12 \text{ mm} \quad (4.79)$$

$$f_{admissivel} = L/250 = 19300/250 = 77,2 \text{ mm} \quad (4.80)$$

Portanto, considerando a aceitabilidade sensorial visual, a flecha total é superior à admissível, logo, não está atendido o estado-limite de deformação excessiva. A Tabela 4.13 apresenta os resultados obtidos pelo programa na verificação.

Tabela 4.13 – Verificação da flecha: aplicação numérica 4

	Programa
Momento de fissuração (kN.cm)	56.605
Posição da linha neutra x_{II} (cm)	47,02
Inércia da seção bruta de concreto I_c (cm ²)	2.500.000
Inércia da seção no estágio II (cm ²)	1.406.817
Rigidez equivalente $(EI)_{eq}$ (kN.cm ²)	$3,967 \times 10^9$
Flecha imediata devida ao carregamento permanente (mm)	27,37
Flecha imediata devida ao carregamento acidental (mm)	15,65
Flecha diferida (mm)	88,47
Flecha total (mm)	104,12
Flecha admissível (mm)	77,2

Fonte: Elaborada pela autora.

Uma possível solução para redução da flecha, de modo que esta se torne inferior à admissível, é o acréscimo de mais uma cordoalha, totalizando cinco cordoalhas CP 190 RB 15,2 mm. Neste caso, seriam obtidos os resultados expostos na Tabela 4.14.

Tabela 4.14 – Verificação da flecha com 5 cordoalhas CP 190 RB 15,2

	Programa
Flecha imediata devida ao carregamento permanente (mm)	12,15
Flecha imediata devida ao carregamento acidental (mm)	9,6
Flecha diferida (mm)	39,28
Flecha total (mm)	48,87
Flecha admissível (mm)	77,2

Fonte: Elaborada pela autora.

4.4.3 Dimensionamento à força cortante

No dimensionamento à força cortante, a distribuição dos estribos é realizada em três faixas, a primeira até a coordenada $x_{f1} = x_{i2} = 6,5$ m, a segunda até $x_{f2} = x_{i3} = 12,8$ m e a terceira até o segundo apoio.

Para consideração do efeito da armadura ativa, utiliza-se a carga equivalente ao esforço de protensão, apresentada em (4.65). Então, são obtidas as forças cortantes solicitantes máximas de cálculo para cada faixa, conforme as equações (4.81) e (4.82).

$$V_{sd,1} = V_{sd,3} = 1,4 \times \frac{(g_1 + g_2 + q_1 + q_2) L}{2} - 0,9 \times \frac{g_{eq} L}{2} = 210,55 \text{ kN} \quad (4.81)$$

$$V_{sd,2} = 210,55 - 1,4 \times (g_1 + g_2 + q_1 + q_2) \times 6,5 + 0,9 \times g_{eq} \times 6,5 = 68,73 \text{ kN} \quad (4.82)$$

A força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas é dada por (4.83), e observa-se que é maior que a força cortante solicitante máxima de cálculo.

$$V_{Rd2} = 0,27 \times \left(1 - \frac{f_{ck}}{250}\right) f_{cd} b_w d = 0,27 \times \left(1 - \frac{30}{250}\right) \times \frac{3}{1,4} \times 30 \times 96 = 1466,33 \text{ kN} \quad (4.83)$$

O momento fletor máximo de cálculo é $M_{sd,max} = 1282,86$ kN.m e o momento fletor que anula σ_p na borda tracionada por M_{sd} , M_0 , é dado por (4.84).

$$M_0 = 0,9 \times P_{inf} \left(e + \frac{W_c}{A_c}\right) = 0,9 \times 705,68 \times \left(0,42 + \frac{0,05}{0,3}\right) = 372,6 \text{ kNm} \quad (4.84)$$

A parcela de força cortante absorvida por mecanismos complementares ao da treliça V_c é dada por (4.85).

$$\begin{aligned} V_c &= 0,009 f_{ck}^{2/3} b_w d \left(1 + \frac{M_0}{M_{sd,max}}\right) = \\ &= 0,009 \times 30^{2/3} \times 30 \times 96 \times \left(1 + \frac{372,6}{1282,86}\right) = 322,94 \text{ kN} \end{aligned} \quad (4.85)$$

Então, calcula-se a área de armadura transversal necessária para a primeira e terceira faixa, ver (4.86).

$$A_{sw1/m} = A_{sw3/m} = \frac{V_{sd} - V_c}{0,9 d f_{ywd}} = \frac{210,55 - 322,94}{0,9 \times 0,96 \times 43,48} = -2,99 \text{ cm}^2/\text{m} \quad (4.86)$$

Observa-se que V_{sd} é menor que V_c , e, portanto, o valor da área de aço necessária é negativo. Logo, é necessário adotar, nas três faixas, a armadura mínima dada por (4.87), em que α é a inclinação dos estribos em relação ao eixo longitudinal e $s = 100$ cm. Portanto, as armaduras transversais podem ser alojadas em 2 ramos de $\phi 5$ mm c/11.

$$A_{swmin} = 0,2 \frac{f_{ct,m}}{f_{yw}} b_w . s . \text{sen } \alpha = 0,2 \times \frac{2,896}{500} \times 30 \times 100 \times \text{sen } 90 = 3,48 \text{ cm}^2/\text{m} \quad (4.87)$$

A Tabela 4.15 apresenta os resultados obtidos pelo programa no dimensionamento à força cortante.

Tabela 4.15 – Dimensionamento à força cortante: aplicação numérica 4

	Programa
Carga equivalente ao esforço de protensão (kN/m)	6,37
Força cortante máxima solicitante de cálculo para 1ª e 3ª faixa (kN)	210,55
Força cortante máxima solicitante de cálculo para 2ª faixa (kN)	68,73
Força cortante resistente de cálculo relativa à ruína das diagonais comprimidas (kN)	1466,33
Parcela de força cortante absorvida por mecanismos complementares ao da treliça (kN)	322,94
Área de armadura transversal necessária para 1ª e 3ª faixa (cm ² /m)	-2,99
Área de armadura transversal mínima (cm ² /m)	3,48

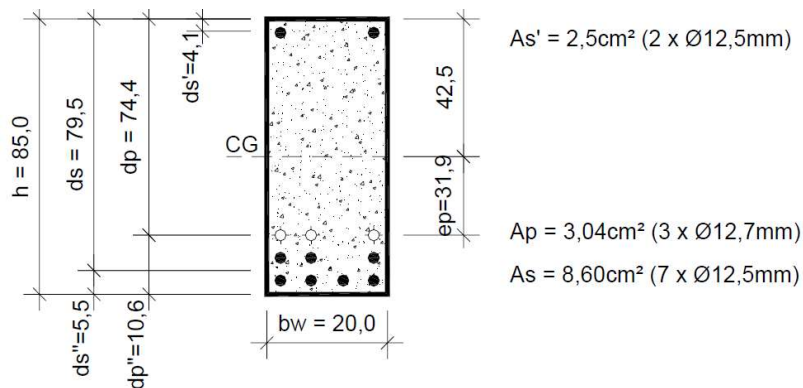
Fonte: Elaborada pela autora.

4.5 Aplicação numérica 5

Zanette (2006) apresenta uma aplicação numérica do dimensionamento de uma viga com protensão parcial, verificação do estado-limite de abertura das fissuras e estado-limite de deformação excessiva, cujos resultados são comparados com os obtidos pelo programa. Trata-se de uma viga de vão $L = 12$ m, com seção retangular conforme a Figura 4-10. O carregamento permanente é $g = 15$ kN/m e o acidental é $q = 5$ kN/m, com $\psi_1 = 0,4$ e $\psi_2 = 0,3$. Classe de agressividade ambiental II, concreto C30 ($E_{cs} = 26071,59$ MPa), cobrimento nominal $c_{nom} = 3,0$ cm para armaduras passivas, aço CA50 ($E_s = 210000$ MPa) e cordoalhas CP190 RB 12,7 ($E_p = 200000$ MPa).

Para o desenvolvimento deste exemplo no programa e obtenção da área de armadura ativa indicada na aplicação numérica, foi adotado balanceamento de 40% do momento devido ao carregamento permanente.

Figura 4-10 – Seção transversal da viga



Fonte: Zanette (2006).

A Tabela 4.16 apresenta, para o meio do vão, os resultados obtidos pelo programa referentes à força inicial e final de protensão, perdas, tensões nas bordas inferior e superior, áreas de armadura ativa e passiva, e, razão entre a posição da linha neutra e altura útil da armadura passiva no estado-limite último (x/d).

Tabela 4.16 – Resultados obtidos para o meio do vão

Viga submetida à protensão parcial	Programa
Força inicial de protensão (kN)	426,55
Perda por atrito (kN)	4,63
Perda por ancoragem (kN)	42,50
Perda por encurtamento (kN)	7,18
Perdas progressivas (kN)	15,54
Força final de protensão (kN)	356,70
Porcentagem total de perdas (%)	16,38
Tensão na borda inferior CF (kN/m ²)	5882,93
Tensão na borda superior CF (kN/m ²)	-10079,39
Armadura ativa estimada $A_{p,est}$ (cm ²)	3,02
Armadura passiva necessária A_s (cm ²)	7,54
ELU: x/d	0,27

Fonte: Elaborada pela autora.

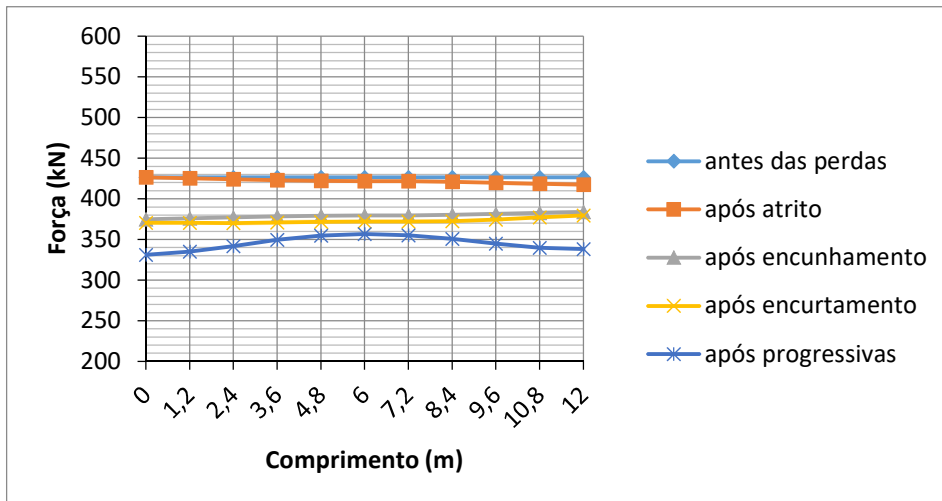
A Tabela 4.17 apresenta os resultados obtidos pelo programa para a excentricidade do cabo, força inicial e final de protensão, perdas e tensões nas bordas inferior e superior nas seções de análise ao longo da viga. Enquanto a Figura 4-11 apresenta a força de protensão ao longo do cabo.

Tabela 4.17 – Perdas de protensão calculadas pelo programa

	Seção 1	Seção 2	Seção 3	Seção 4	Seção 5	Seção 6	Seção 7	Seção 8	Seção 9	Seção 10	Seção 11
Posição (m)	0	1,2	2,4	3,6	4,8	6,0	7,2	8,4	9,6	10,8	12,0
Excentricidade (m)	0	0,16	0,27	0,32	0,32	0,32	0,32	0,32	0,27	0,16	0
Força inicial de protensão (kN)	426,55	426,55	426,55	426,55	426,55	426,55	426,55	426,55	426,55	426,55	426,55
Perda por atrito (kN)	0	1,26	2,52	3,78	4,37	4,63	4,88	5,47	6,73	7,97	9,21
Perda por ancoragem (kN)	51,76	49,25	46,72	44,19	43,01	42,50	42,0	40,82	38,31	35,81	33,35
Perda por encurtamento (kN)	4,34	5,45	7,11	7,8	7,36	7,18	7,38	7,85	7,22	5,56	4,48
Perdas progressivas (kN)	39,12	35,67	28,25	21,17	16,92	15,54	17,06	21,62	29,34	37,32	41,30
Força final de protensão (kN)	331,30	334,92	341,95	349,60	354,89	356,70	355,24	350,79	344,96	339,88	338,21
% total de perdas	22,33	21,48	19,83	18,04	16,80	16,38	16,72	17,76	19,13	20,32	20,71
Tensão na borda inferior (kN/m ²)	-1948,8	341,5	2351,7	4032,0	5409,4	5882,9	5402,6	4009,6	2264,4	278,78	-1989,5
Tensão na borda superior (kN/m ²)	-1948,8	-4281,8	-6338,6	-8145,0	-9584,5	-10079	-9581,9	-8136,5	-6322,8	-4277,4	-1989,5

Fonte: Elaborada pela autora.

Figura 4-11 – Força ao longo do cabo (programa)



Fonte: Elaborada pela autora.

A Tabela 4.18 apresenta, para as seções de análise no estado-limite último, os resultados obtidos pelo programa para a força de protensão após as perdas, o momento solicitante de cálculo M_{Sd} , o momento resistente de cálculo M_{Rd} , da razão entre a posição da linha neutra e a altura útil da armadura passiva x/d , a área de armadura passiva necessária A_s e a mínima $A_{s,min}$. Não foi considerada a decalagem.

Tabela 4.18 – Estado-limite último (programa)

	Seção	Seção	Seção	Seção	Seção	Seção	Seção	Seção	Seção	Seção	Seção
	1	2	3	4	5	6	7	8	9	10	11
Posição (m)	0	1,2	2,4	3,6	4,8	6,0	7,2	8,4	9,6	10,8	12,0
Força final de protensão (kN)	331,30	334,92	341,95	349,60	354,89	356,70	355,24	350,79	344,96	339,88	338,21
$M_{Sd} = M_{Rd}$ (kN.m)	0	181,4	322,6	423,4	483,8	504	483,8	423,4	322,6	181,4	0
x/d	0,018	0,102	0,168	0,215	0,256	0,27	0,256	0,215	0,168	0,102	0,019
Armadura passiva necessária A_s (cm ²)	-8,08	-2,67	1,64	4,79	6,83	7,54	6,83	4,76	1,58	-2,77	-8,21
Armadura passiva mínima $A_{s,min}$ (cm ²)	1,28	1,28	1,28	1,28	1,28	1,28	1,28	1,28	1,28	1,28	1,28

Fonte: Elaborada pela autora.

4.5.1 Protensão parcial: verificação do estado-limite de abertura das fissuras

A Tabela 4.19 apresenta os resultados da verificação da abertura das fissuras obtidos pelo programa e compara com os resultados obtidos por Zanette (2006).

Tabela 4.19 – Protensão parcial: aplicação numérica 5

	Zanette (2006)	Programa
Área de armadura ativa A_p (cm ²)	3,04	3,02
Alojamento da armadura ativa	3 ϕ 12,7	3 ϕ 12,7
Alojamento da armadura passiva	7 ϕ 12,5	7 ϕ 12,5
Área de armadura passiva A_s (cm ²)	2,5	2,5
Momento fletor na combinação frequente M_{CF} (kN.m)	306	306
Posição da linha neutra x (cm)	25,1	35,93
Tensão na armadura passiva σ_s (kN/cm ²)	22,9	12,38
Área de envolvimento das armaduras A_{cri} (cm ²)	335	335
Abertura característica das fissuras w (mm)	0,010	0,034

Fonte: Elaborada pela autora.

Observa-se grande diferença nos valores da posição da linha neutra e tensão na armadura, e consequentemente, no valor característico da abertura das fissuras. Tal fato se justifica pela diferença entre os métodos de análise utilizados.

O programa realiza o equilíbrio de forças e momentos na seção, e tem como solução o par x e σ_s , que atende simultaneamente às equações (4.88) e (4.89), para seções retangulares com protensão não aderente.

$$\sigma_s = \frac{\varepsilon_{pré} E_p A_p}{\frac{b_c \cdot x}{2\alpha_e} \frac{x}{d-x} + A_s' \frac{x-d'}{d-x} - A_s} \quad (4.88)$$

$$\sigma_s = \frac{M_{CF} - \varepsilon_{pré} E_p A_p (d_p - y_{cg})}{\frac{b_c \cdot x}{2\alpha_e} \frac{x}{d-x} (y_{cg} - x/3) + A_s' \frac{x-d'}{d-x} (y_{cg} - d') + A_s (d - y_{cg})} \quad (4.89)$$

Já Zanette (2006), considera, inicialmente, a seção homogeneizada no estágio II como se a estrutura fosse de concreto armado. Explica que a linha neutra coincide com o centro de gravidade da figura da área de concreto comprimido e áreas equivalentes tracionadas e comprimidas. Assim, iguala o momento estático da seção homogeneizada em relação à linha neutra a zero e calcula o momento de inércia I_{II} , a posição da linha neutra x e a área da seção homogeneizada A_{II} , conforme as equações (4.90) a (4.97).

$$Q = b_w x_{II} \frac{x_{II}}{2} + (\alpha_e - 1) A_s' (x_{II} - d') - \alpha_e A_s (d - x_{II}) = 0 \quad (4.90)$$

$$\frac{b_w}{2} x_{II}^2 + [\alpha_e A_s + (\alpha_e - 1) A_s'] x_{II} - [\alpha_e A_s d + (\alpha_e - 1) A_s' d'] = 0 \quad (4.91)$$

$$x = \frac{-B + \sqrt{B^2 - 4AC}}{2A} = \frac{-164 + \sqrt{164^2 - 4 \times 10 \times (-10399)}}{2 \times 10} = 25,07 \text{ cm} \quad (4.92)$$

$$A = b_c/2 = \frac{20}{2} = 10 \text{ cm} \quad (4.93)$$

$$B = \alpha_e A_s + (\alpha_e - 1) A_s' = 15 \times 8,6 + (15 - 1) \times 2,5 = 164 \text{ cm}^2 \quad (4.94)$$

$$C = -\alpha_e A_s d - (\alpha_e - 1) A_s' d' = -15 \times 8,6 \times 79,5 - (15 - 1) \times 2,5 \times 4,1 = -10399 \text{ cm}^3 \quad (4.95)$$

$$I_{II} = \frac{b_c x^3}{3} + \alpha_e A_s (d - x)^2 + (\alpha_e - 1) A_s' (x - d')^2 =$$

$$= \frac{20 \times 25,07^3}{3} + 15 \times 8,6 \times (79,5 - 25,07)^2 + (15 - 1) \times$$

$$\times 2,5 \times (25,07 - 4,1)^2 = 502613,66 \text{ cm}^4 \quad (4.96)$$

$$A_{II} = b_c x + \alpha_e A_s + (\alpha_e - 1) A_s' = 20 \times 25,07 + 15 \times 8,6 + (15 - 1) \times 2,5 = 665,4 \text{ cm}^2 \quad (4.97)$$

Então, considera o efeito da protensão como um carregamento externo equivalente, decomposto em um esforço normal, aplicado no centro de gravidade da seção bruta de concreto, e um momento fletor considerado nas combinações de serviço, ver (4.98) e (4.99). Sendo a força de protensão após as perdas, utilizada por Zanette (2006), igual a 360 kN.

$$M_{d, serv} = M_{g,k} + M_{p,k} + \Psi_1 M_{q,k} = \frac{15 \times 12^2}{8} - 360 \times 0,319 + 0,4 \times \frac{5 \times 12^2}{8} = \quad (4.98)$$

$$= 270 - 114,84 + 36 = 191,16 \text{ kNm}$$

$$\sigma_s = \alpha_e \frac{P_{inf}}{A_{II}} + \alpha_e \frac{M_{d, serv} \times (d - x)}{I_{II}} = \quad (4.99)$$

$$= -15 \times \frac{360}{665,4} + 15 \times \frac{19116 \times (79,5 - 25,07)}{502613,66} =$$

$$= 22,9 \text{ kN/cm}^2$$

Outra diferença ocorre no valor de α_e . O programa adota α_e igual à razão entre o módulo de elasticidade do aço e o módulo de elasticidade secante do concreto. Já Zanette (2006) utiliza α_e igual 15, valor permitido pela NBR 6118:2014 para o cálculo no estágio II.

Observa-se que ao considerar o valor da posição da linha neutra encontrado pelo programa nas equações de Zanette (2006), o valor da tensão na armadura ativa é relativamente próximo ao encontrado pelo programa.

4.5.2 Verificação do estado-limite de deformação excessiva

Inicialmente, o programa calcula o carregamento equivalente à protensão, apresentado em (4.100)

$$g_{eq} = \frac{8 \times P_{inf} \times e}{L^2} = \frac{8 \times 356,70 \times 0,319}{12^2} = 6,32 \text{ kN/m} \quad (4.100)$$

Então, pela equação da linha elástica, o programa calcula o deslocamento, na seção mais solicitada da viga, considerando a carga permanente g e o alívio dos cabos g_{eq} , ver (4.101) e (4.102). Calcula, também, a flecha devida ao carregamento acidental, ver (4.103) e (4.104). Considera-se a combinação quase permanente.

$$g_{total} = g - g_{eq} = 4,25 + 10,75 - 6,32 = 8,68 \text{ kN/m} \quad (4.101)$$

$$f_{elástica,g} = 8,78 \text{ mm} \quad (4.102)$$

$$q_{total} = \Psi_2 \times q = 0,3 \times 5 = 1,5 \text{ kN/m} \quad (4.103)$$

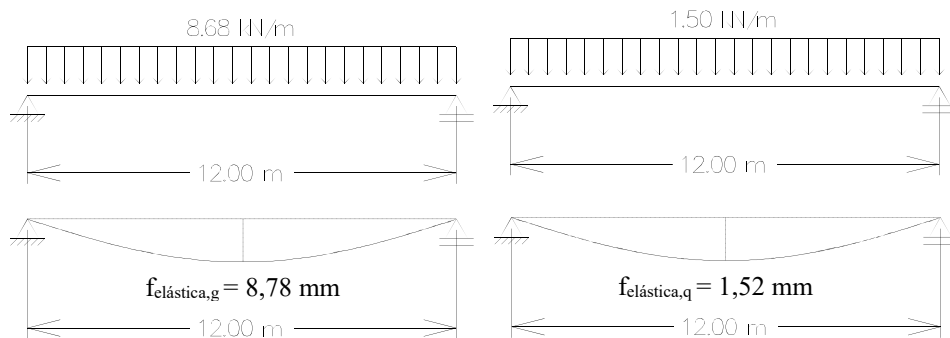
$$f_{elástica,q} = 1,52 \text{ mm} \quad (4.104)$$

As flechas elásticas calculadas pelo programa foram comparadas com as analíticas, (4.105) e (4.106), considerando a seção bruta de concreto, ver Figura 4-12. Como se observa, as flechas obtidas são as mesmas.

$$f_{elástica,g} = \frac{5 \times g_{total} \times L^4}{384E_{cs}I_c} = \frac{5 \times 8,68 \times 12^4}{384 \times 26071590 \times 0,010235} = 0,00878 \text{ m} \quad (4.105)$$

$$f_{elástica,q} = \frac{5 \times q_{total} \times L^4}{384E_{cs}I_c} = \frac{5 \times 1,5 \times 12^4}{384 \times 26071590 \times 0,010235} = 0,00152 \text{ m} \quad (4.106)$$

Figura 4-12 – Deslocamentos



Fonte: Elaborada pela autora.

Como se trata de protensão parcial e, portanto, seção fissurada, deve-se calcular a rigidez equivalente para obtenção das flechas imediatas, conforme a equação (4.107). Para tanto, calcula-se o momento de fissuração pelas expressões (4.108) e (4.109). Para o cálculo do momento de inércia no estágio II, utiliza-se a equação (4.110). Considera-se a combinação quase permanente para a determinação da posição da linha neutra, $x_{II} = 37,03 \text{ cm}$.

$$\begin{aligned} (EI)_{eq} &= E_{cs} \left\{ \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 I_c + \left[1 - \left(\frac{M_r - M_0}{M_{CQP} - M_0} \right)^3 \right] I_{II} \right\} = \\ &= 2607,16 \times \left\{ \left(\frac{23792,09 - 16467,65}{30600 - 16467,65} \right)^3 \times 1023500 + \right. \\ &\quad \left. + \left[1 - \left(\frac{23792,09 - 16467,65}{30600 - 16467,65} \right)^3 \right] \times 482235,1 \right\} \quad (4.107) \\ &= 2607,16 \times 574038 = 1,454 \times 10^9 \text{ kN.cm}^2 \end{aligned}$$

$$M_0 = P_{inf} \left(e + \frac{W_c}{A_c} \right) = 356,7 \left(0,319 + \frac{0,02408}{0,17} \right) = 164,68 \text{ kNm} = 16467,65 \text{ kNcm} \quad (4.108)$$

$$M_r = M_0 + W_c f_{ct} = 164,68 + 0,02408 \times 3041,29 = 237,92 \text{ kNm} = 23792,09 \text{ kNcm} \quad (4.109)$$

$$I_{II} = \frac{b_c x^3}{3} + \frac{E_s}{E_{cs}} A_s (d - x)^2 + \left(\frac{E_s}{E_{cs}} - 1 \right) A_s' (x - d')^2 =$$

$$= \frac{20 \times 37,03^3}{3} + \frac{21000}{2607,16} \times 8,6 \times (79,5 - 37,03)^2 +$$

$$+ \left(\frac{21000}{2607,16} - 1 \right) \times 2,5 \times (37,03 - 4,1)^2 = 482235,1 \text{ cm}^4 \quad (4.110)$$

Então, as flechas imediatas, na seção mais solicitada, podem ser calculadas por (4.111) e (4.112).

$$f_{\text{imediata},g} = f_{\text{elástica},g} \frac{E_{cs} I_c}{(EI)_{eq}} = 8,78 \text{ mm} \times \frac{2607,16 \times 1023500}{2607,16 \times 574038} = 15,66 \text{ mm} \quad (4.111)$$

$$f_{\text{imediata},q} = f_{\text{elástica},q} \frac{E_{cs} I_c}{(EI)_{eq}} = 1,52 \text{ mm} \times \frac{2607,16 \times 1023500}{2607,16 \times 574038} = 2,7 \text{ mm} \quad (4.112)$$

A flecha diferida, referente ao carregamento permanente, é dada por (4.113).

$$f_{\text{diferida},g} = (1 + \varphi_{\infty}) f_{\text{imediata},g} = (1 + 2,276) \times 15,66 = 51,3 \text{ mm} \quad (4.113)$$

A flecha total é a soma da flecha diferida, devida ao carregamento permanente, com a flecha imediata devida à carga acidental, o programa, então, compara o valor calculado com o limite estabelecido pela NBR 6118:2014, considerando a aceitabilidade sensorial visual. Ver (4.114) e (4.115).

$$f_{\text{total}} = f_{\text{diferida},g} + f_{\text{imediata},q} = 51,3 + 2,7 = 54 \text{ mm} \quad (4.114)$$

$$f_{\text{admissível}} = L/250 = 12000/250 = 48 \text{ mm} \quad (4.115)$$

Portanto, considerando a aceitabilidade sensorial visual, a flecha total é superior à admissível, logo, não está atendido o estado-limite de deformação excessiva. A Tabela 4.20 apresenta os resultados obtidos pelo programa na verificação.

Tabela 4.20 – Verificação da flecha: aplicação numérica 5

	Programa
Momento de fissuração (kN.cm)	23.792
Posição da linha neutra x_{II} (cm)	37,03
Inércia da seção bruta de concreto I_c (cm ²)	1.023.500
Inércia da seção no estágio II (cm ²)	482.235
Rigidez equivalente $(EI)_{eq}$ (kN.cm ²)	1,454 x 10 ⁹
Flecha imediata devida ao carregamento permanente (mm)	15,66
Flecha imediata devida ao carregamento acidental (mm)	2,7
Flecha diferida (mm)	51,3
Flecha total (mm)	54
Flecha admissível (mm)	48

Fonte: Elaborada pela autora.

Uma possível solução para redução da flecha, de modo que esta se torne inferior à admissível, é o acréscimo de mais uma cordoalha, totalizando quatro cordoalhas CP 190 RB 12,7 mm. Neste caso, seriam obtidos os resultados expostos na Tabela 4.21.

Tabela 4.21 – Verificação da flecha com 4 cordoalhas CP 190 RB 12,7

	Resultados
Flecha imediata devida ao carregamento permanente (mm)	6,71
Flecha imediata devida ao carregamento acidental (mm)	1,52
Flecha diferida (mm)	21,98
Flecha total (mm)	23,49
Flecha admissível (mm)	48

Fonte: Elaborada pela autora.

4.5.3 Dimensionamento à força cortante

No dimensionamento à força cortante, a distribuição dos estribos é realizada em três faixas, a primeira até a coordenada $x_{f1} = x_{i2} = 4$ m, a segunda até $x_{f2} = x_{i3} = 8$ m e a terceira até o segundo apoio.

Para consideração do efeito da armadura ativa, utiliza-se a carga equivalente ao esforço de protensão, apresentada em (4.100). Então, são obtidas as forças cortantes solicitantes máximas de cálculo para cada faixa, conforme as equações (4.116) e (4.117).

$$V_{sd,1} = V_{sd,3} = 1,4 \times \frac{(g + q) L}{2} - 0,9 \times \frac{g_{eq} L}{2} = 133,87 \text{ kN} \quad (4.116)$$

$$V_{sd,2} = 133,87 - 1,4 \times (g + q) \times 4 + 0,9 \times g_{eq} \times 4 = 44,62 \text{ kN} \quad (4.117)$$

A força cortante resistente de cálculo, relativa à ruína das diagonais comprimidas é dada por (4.118), e observa-se que é maior que a força cortante solicitante máxima de cálculo.

$$V_{Rd2} = 0,27 \times \left(1 - \frac{f_{ck}}{250}\right) f_{cd} b_w d = 0,27 \times \left(1 - \frac{30}{250}\right) \times \frac{3}{1,4} \times 20 \times 79,5 = 809,36 \text{ kN} \quad (4.118)$$

O momento fletor máximo de cálculo é $M_{sd,max} = 504 \text{ kN.m}$ e o momento fletor que anula σ_p na borda tracionada por M_{sd} , M_0 , é dado por (4.119).

$$M_0 = 0,9 \times P_{inf} \left(e + \frac{W_c}{A_c}\right) = 0,9 \times 356,7 \times \left(0,319 + \frac{0,02408}{0,17}\right) = 148,21 \text{ kNm} \quad (4.119)$$

A parcela de força cortante absorvida por mecanismos complementares ao da treliça V_c é dada por (4.120).

$$\begin{aligned} V_c &= 0,009 f_{ck}^{2/3} b_w d \left(1 + \frac{M_0}{M_{sd,max}}\right) = \\ &= 0,009 \times 30^{2/3} \times 20 \times 79,5 \times \left(1 + \frac{148,21}{504}\right) = 178,75 \text{ kN} \end{aligned} \quad (4.120)$$

Então, calcula-se a área de armadura transversal necessária para a primeira e terceira faixa, ver (4.121).

$$A_{sw1/m} = A_{sw3/m} = \frac{V_{sd} - V_c}{0,9 d f_{ywd}} = \frac{133,87 - 178,75}{0,9 \times 0,795 \times 43,48} = -1,44 \text{ cm}^2/m \quad (4.121)$$

Observa-se que V_{sd} é menor que V_c , e, portanto, o valor da área de aço necessária é negativo. Logo, é necessário adotar, nas três faixas, a armadura mínima dada por (4.122), em que α é a inclinação dos estribos em relação ao eixo longitudinal e $s = 100 \text{ cm}$. Portanto, as armaduras transversais podem ser alojadas em 2 ramos de $\phi 5 \text{ mm c}/16$.

$$A_{swmin} = 0,2 \frac{f_{ct,m}}{f_{ywk}} b_w \cdot s \cdot \text{sen } \alpha = 0,2 \times \frac{2,896}{500} \times 30 \times 100 \times \text{sen } 90 = 3,48 \text{ cm}^2/m \quad (4.122)$$

A Tabela 4.22 apresenta os resultados obtidos pelo programa no dimensionamento à força cortante.

Tabela 4.22 – Dimensionamento à força cortante: aplicação numérica 5

	Resultados
Carga equivalente ao esforço de protensão (kN/m)	6,37
Força cortante máxima solicitante de cálculo para 1ª e 3ª faixa (kN)	210,55
Força cortante máxima solicitante de cálculo para 2ª faixa (kN)	68,73
Força cortante resistente de cálculo relativa à ruína das diagonais comprimidas (kN)	1466,33
Parcela de força cortante absorvida por mecanismos complementares ao da treliça (kN)	322,94
Área de armadura transversal necessária para 1ª e 3ª faixa (cm ² /m)	-2,99
Área de armadura transversal mínima (cm ² /m)	3,48

Fonte: Elaborada pela autora.

5 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

5.1 Conclusões

Neste trabalho foi elaborada uma ferramenta computacional para o dimensionamento de vigas protendidas biapoiadas, parcialmente protendidas, conforme a NBR 6118:2014, expandindo o programa iniciado por Dominicini e Coelho (2014). O programa foi desenvolvido utilizando o *Microsoft Excel* e o *Microsoft Visual Basic for Applications*.

O programa abrange vigas protendidas com armaduras ativas aderentes ou não aderentes, no caso de pós-tração, para concretos até a classe C50. Além do nível de protensão parcial, contempla, também, os níveis de protensão limitada e completa.

A ferramenta calcula os esforços solicitantes ao longo da viga, determina a força de protensão necessária, o alojamento e o traçado dos cabos. Em seguida, calcula as perdas de protensão imediatas e progressivas. Verifica os estados-limites de serviço, conforme cada nível de protensão, inclusive o estado-limite de deformação excessiva. Executa o dimensionamento no estado-limite último à flexão e cisalhamento. Para o caso de protensão parcial é realizada a verificação do estado-limite de abertura das fissuras, após o alojamento das armaduras passivas.

Foram desenvolvidos exemplos presentes na literatura. A primeira aplicação numérica trata de uma viga de seção retangular, parcialmente protendida, com protensão aderente, apresentada em França (2001), os resultados obtidos foram satisfatórios. Foi, ainda, desenvolvido no programa, um segundo exemplo apresentado em França (2001), considerando seção T com protensão não-aderente, obtendo de maneira semelhante resultados próximos.

No terceiro exemplo, foi desenvolvida a aplicação numérica apresentada em Nilson (1976), trata-se de uma viga parcialmente protendida, com protensão aderente, em que são calculadas as tensões atuantes no concreto e armaduras ativas e passivas, além da posição da linha neutra no estágio II. Os resultados obtidos foram relativamente próximos, deve-se considerar que houve pequenas diferenças nas alturas úteis adotadas pelo programa no alojamento e aproximações nas transformações de unidades.

No quarto exemplo, foi realizado o dimensionamento de uma viga retangular com protensão não-aderente apresentada em Cholfé e Bonilha (2013). Os resultados apresentados pelo programa para o dimensionamento das armaduras ativas e passivas foram satisfatórios. Houve divergência na verificação da abertura das fissuras, por conta de inconsistências nas equações desenvolvidas em Cholfé e Bonilha (2013), entretanto, realizando ajustes nessas equações, os resultados são os mesmos apresentados pelo programa.

Por fim, foi realizado um exemplo desenvolvido em Zanette (2006), de uma viga retangular com protensão não-aderente e armadura dupla. Os resultados apresentados pelo programa para o dimensionamento das armaduras ativas e passivas foram satisfatórios. Houve divergência na verificação da abertura das fissuras, uma vez que o programa realiza o equilíbrio de forças e momentos na seção, enquanto Zanette (2006) inicialmente considera a seção homogeneizada no estágio II como se a estrutura fosse de concreto armado, e posteriormente inclui o efeito da protensão somando a tensão devido à força normal aplicada e considerando o alívio no momento de serviço.

Nas aplicações numéricas foram desenvolvidas as verificações de estado-limite de deformação excessiva e dimensionamento à força cortante. Os resultados obtidos pelo programa foram comparados com cálculos manuais, baseados na literatura.

Diante disso, pode-se afirmar que o programa é capaz de realizar o dimensionamento de vigas protendidas biapoiadas de maneira satisfatória no nível de protensão parcial, verificando o estado-limite de aberturas das fissuras, e estado-limite de deformação excessiva, área ainda pouco explorada pelos programas acadêmicos. A ferramenta também apresenta bons resultados para o dimensionamento à força cortante, além de apresentar interface gráfica com o usuário, o que facilita e amplia sua utilização. A geração de relatórios e memórias de cálculo torna a ferramenta mais didática, possibilitando sua adoção como recurso auxiliar no meio acadêmico tanto no nível de graduação e quanto de pós-graduação.

5.2 Sugestões para trabalhos futuros

Como sugestão para trabalhos futuros, indica-se:

- Inclusão da consideração de concretos de classes superiores a C50;
- Inclusão de cargas móveis;

- Implementação de seções I na verificação da abertura de fissuras;
- Inclusão da consideração da decalagem para a armação longitudinal;
- Inclusão do dimensionamento da armadura de fretagem;
- Inclusão da consideração dos momentos nos apoios, em casos de excentricidade inicial no cabo resultante;
- Inclusão da geração de detalhamento e quantitativo das armações passivas longitudinais e transversais;
- Inclusão da geração do detalhamento do perfil dos cabos e quantitativos;
- Expansão para vigas isostáticas com balanços;
- Expansão para vigas contínuas;
- Expansão para lajes.

REFERÊNCIAS

AALAMI, B. O. Critical Milestones in Development of Post-Tensioned Buildings. **Concrete International**. 2007; p. 52-56.

AALAMI, B. O. **Post-Tensioned Buildings: Design and Construction**. International Edition, 2014.

AMERICAN CONCRETE INSTITUTE. **Building Code Requirements for Structural Concrete and Commentary**. ACI 318, Farmington Hills, 2011.

ASSIS, W. S.; BITTENCOURT, T. N.; NORONHA, M. Desenvolvimento de recursos multimídia para o ensino de engenharia de estruturas. **Instituto Brasileiro do Concreto**. São Paulo, 2002.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Projeto de Estruturas de Concreto – Procedimento**. NBR 6118. Rio de Janeiro, 2014.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Projeto e execução de obras de concreto armado**. NBR 6118. Rio de Janeiro, 1978.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Fios de aço para estruturas de concreto protendido – Especificação**. NBR 7482. Rio de Janeiro, 2008.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Cordoalhas de aço para estruturas de concreto protendido – Especificação**. NBR 7483. Rio de Janeiro, 2008.

CARNEIRO, A. L. **Análise e dimensionamento de lajes lisas protendidas sem aderência**. Dissertação de Mestrado, Universidade Federal do Espírito Santo, Vitória, 2015.

CARVALHO, R. C. **Estruturas em Concreto Protendido: cálculo e detalhamento**. São Paulo: PINI, 2012.

CHOLFE, L.; BONILHA, L. **Concreto Protendido: Teoria e prática**. São Paulo: PINI, 2013.

DOMINICINI, W. K.; COELHO, L. H. Desenvolvimento de software educacional para análise e dimensionamento de estruturas em concreto protendido. **Congresso Brasileiro de Ensino de Engenharia**. Juiz de Fora, 2014.

EMERICK, A. A. **Projeto e Execução de Lajes Protendidas**. Rio de Janeiro, Ed. Interciência, 2005.

FERREIRA, W. B. **Estudo de desempenho e critérios de abordagem para lajes lisas protendidas**. Dissertação de Mestrado, Universidade Federal do Espírito Santo, Vitória, 2013.

FRANÇA, R. L. S. **Concreto Protendido**. Notas de aula. Universidade de São Paulo, São Paulo, 2001.

HANAI, J. B. **Fundamentos do Concreto Protendido**. E-book. São Carlos: EESC-USP, 2005.

HERNANDEZ, S.; MARCOS, D.; FONTAN, A. N.; DIAZ, J. VTOP – an improved software for design optimization of prestressed concrete beams. **WIT Transactions on The Built Environment**. v. 91, p. 151-164, 2007.

KLEIN, R.; LORIGGIO, D. D. Dimensionamento por computador de vigas simplesmente apoiadas de concreto protendido pós tracionadas. **Anais do VI Simpósio EPUSP sobre Estruturas de Concreto**. São Paulo, 2006; p. 1132-1144.

ISHITANI, H. **Concreto Protendido**. Notas de aula. Universidade de São Paulo – Escola Politécnica, São Paulo, 1998.

JOST, D. T. **Análise de peças fletidas com protensão não aderente pelo método dos elementos finitos**. Dissertação de Mestrado, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2006.

LABADAN, R. S. Design of post-tensioned prestressed concrete beam using Excel spreadsheet with Visual Basic Applications. **International Journal of Advances in Mechanical and Civil Engineering**, v. 3, p. 10-16, 2016.

LAZZARI, P. M.; CAMPOS FILHO, A.; GASTAL, F. P. S. L; BARBIERI, R.A.; SCHWINGEL, R. C. Automation of the evaluation of bonded and unbonded prestressed concrete beams, according to brazilian and french code specification. **Ibracon Structures and Materials Journal**. São Paulo, 2013; v. 6; p. 13-54.

LEITE JÚNIOR, J. C. C. **Dimensionamento à punção em apoios internos de lajes protendidas sem aderência**. Dissertação de Mestrado, Universidade Federal do Espírito Santo, Vitória, 2015.

LEONHARDT, F. **Construções de Concreto: Concreto Protendido**. Vol. 5. Rio de Janeiro, Ed. Interciência, 1983.

LIN, T. Y. **Design of Prestressed Concrete Structures**. 3.Ed. New York – USA, John Wiley & Sons, Inc. 1981.

LINDQUIST, M. **Aplicativo Windows para análise e dimensionamento de pórticos planos em concreto armado**. Dissertação de Mestrado, Escola de Engenharia de São Carlos – USP, São Carlos, 2002.

LUCHI, L. R. **Concreto Protendido**. Notas de aula. Universidade Federal do Espírito Santo, Vitória, 2016.

MELLO, A. L. V. **Cálculo de lajes lisas com protensão parcial e limitada**. Dissertação de Mestrado, Universidade Federal de São Carlos, São Carlos, 2005.

MILANI, A. C. **Análise de lajes planas protendidas pelo método dos elementos finitos**. Dissertação de Mestrado, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2006.

MONTEIRO, J. M. A.; BONO, G. F. F.; BONO, G. Programa Computacional para dimensionamento de seções transversais de concreto protendido. **Mecánica Computacional Vol XXXIV**. Córdoba, 2016; p. 601-614.

NAAMAN, A. E. **Prestressed Concrete Analysis and Design: Fundamentals**. 3.Ed. Michigan: Techno Press 3000, 2012.

NACHT, P. K. K. **Ferramenta gráfico-interativa de verificação de tensões no estado limite se serviço de vigas protendidas com pós-tração aderente**. Dissertação de Mestrado, Pontificia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2015.

NAWY, E. G. **Prestressed Concrete – A Fundamental Approach**. 5.Ed. New Jersey: Prentice Hall, 2010.

NÓBREGA, R. J. C.; HOROWITZ, B. Traçado dos Cabos em Vigas Contínuas Protendidas. **Anais do V Simpósio EPUSP sobre Estruturas de Concreto**. São Paulo, 2003.

NILSON, A. H. Flexural stresses after cracking in partially prestressed beams. **PCI JOURNAL**, p.72-81, July-August 1976.

PFEIL, W. **Concreto Protendido: Processos construtivos Perdas de protensão**. Vol. 2. 2.Ed. Rio de Janeiro: Livros Técnicos e Científicos Editora LTDA, 1983.

SALES, R. P. **Uma ferramenta 3D, via web, para dimensionamento de seções retangulares de concreto armado com esboço da armadura**. Dissertação de Mestrado, Universidade Federal de Minas Gerais, Belo Horizonte, 2010.

SILVA, L. M. **Programa computacional para análise e verificação de seções de concreto armado e protendido com flexão oblíqua composta**. Dissertação de Mestrado, Universidade Federal de São Carlos, São Carlos, 2015.

TQS INFORMÁTICA LTDA. **Manual – Lajes Protendidas**. São Paulo, 2015.

VELA, J. M.; CEA, A. A.; BERNAT, A. R. M. **Hormigón armado y Pretensado I**. Barcelona: Edicions UPC, 1993.

VENTRI, D. A. B.; LINDENBERG NETO, H. Utilizando animações para introduzir conceitos fundamentais da mecânica das estruturas. **XXIX Congresso Brasileiro de Ensino de Engenharia**. Porto Alegre, 2001.

VERÍSSIMO, G. S; CÉSAR, K. M. L. **Concreto Protendido – Fundamentos Básicos**. 4. Ed. Universidade Federal de Viçosa, Departamento de Engenharia Civil, 1998.

ZANETTE, D. S. **Projeto de vigas de pequeno porte parcialmente protendidas com monocordoalhas engraxadas**. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Florianópolis, 2006.

APÊNDICE 1

O Apêndice 1 apresenta o código do programa.

Módulo 1

```
Sub Entrada1(control As IRibbonControl)
    Data1.Show
End Sub
```

```
Sub Entrada2(control As IRibbonControl)
    Data2.Show
End Sub
```

```
Sub Entrada3(control As IRibbonControl)
    Data3.Show
End Sub
```

```
Sub Entrada4(control As IRibbonControl)
    Data4.Show
End Sub
```

```
Sub Entrada5(control As IRibbonControl)
    Data5.Show
End Sub
```

```
Sub Entrada6(control As IRibbonControl)
    Data6.Show
End Sub
```

```
Sub Entrada7(control As IRibbonControl)
    Data7.Show
End Sub
```

```
Sub Entrada8(control As IRibbonControl)
    Data8.Show
End Sub
```

```
Sub Entrada9(control As IRibbonControl)
    Data9.Show
End Sub
```

```
Sub BtSalvar(control As IRibbonControl)
    SalvarProjeto
End Sub
```

```
Sub BtFechar(control As IRibbonControl)
    Fechar
End Sub
```

```
Sub BtSobre(control As IRibbonControl)
    Sobre.Show
End Sub
```

```
Sub BtManual(control As IRibbonControl)
    Manual.Show
End Sub
```

```
Sub BtNovo(control As IRibbonControl)
    NovoProjeto
End Sub
```

```
Sub BtImprimir(control As IRibbonControl)
    Imprimir
End Sub
```

```
Sub BtRelatorio(control As IRibbonControl)
    Relatorio
End Sub
```

Data 1

```
Private TESTE As Integer 'Testa se o formulário foi
    fechado pelo X
```

```
Private Eci As Variant 'Modulo de elasticidade
    tangente
```

```
Private Ecs As Variant 'Modulo de elasticidade
    tangente
```

```
Private Sub Bt_AddArea_Click()
    AdicionarAreaElementar.Show
End Sub
```

```
Private Sub Bt_InfoArea_Click()
    MsgBox "Para cálculo automático, a seção
    transversal será subdividida em seções elementares
    retangulares ou triangulares. Informar base, altura e
    coordenada 'y' do centro de gravidade de cada seção
    elementar.", vbInformation, "Informações"
End Sub
```

```
Private Sub Bt_LimpArea_Click()
    ult = Ultima_B(1, 1)
    If ult > 1 Then Plan2.Range("A2:I" & ult).Delete
    Shift:=xlUp
End Sub
```

```
Private Sub btOK_1_Click()
    ActiveSheet.Range("A1:K4").Select
    With Selection
        .HorizontalAlignment = xlCenter
        .VerticalAlignment = xlCenter
        .MergeCells = True
    End With
```

```
With Selection.Font
    .Size = 24
    .Bold = True
End With
```

```

ActiveSheet.Range("A1") = "PROJETO"
    L = Plan3.Range("B12")
If TextBox1 = "" Or TextBox2 = "" Or TextBox3 =
"" Or TextBox4 = "" Or TextBox10 = "" Or
TextBox11 = "" Then
    MsgBox "Preencha todos os dados.",
vbExclamation, "Dados incompletos"
    Exit Sub
End If

If TextBox3 < 10 Then
    MsgBox "Número de seções deve ser no
mínimo 10.", vbOKOnly, "Dado Incorreto"
    Exit Sub
End If

Plan3.Range("B6") = Cdbl(TextBox1)
Plan3.Range("B12") = Cdbl(TextBox2)
Plan3.Range("C13") = Cdbl(TextBox3)
Plan3.Range("B16") = Cdbl(TextBox4)
Plan3.Range("B17") = Cdbl(TextBox10)
Plan3.Range("E18") = Cdbl(Eci)
Plan3.Range("E19") = Cdbl(Ecs)
Plan3.Range("E20") = Cdbl(TextBox11)
Plan3.Range("A15:K21").Copy
Destination:=Plan1.Range("A15")

If CheckBox1 Then
    Plan3.Range("B7") = TextBox5
    Plan3.Range("D8") = TextBox6
    Plan3.Range("D9") = TextBox7
    Plan3.Range("E10") = TextBox8
    Plan3.Range("E11") = TextBox9
    Plan3.Range("A5:K14").Copy
Destination:=Plan1.Range("A5")
    SecaoConhecida = 1
Else
    PropGeo
    SecaoConhecida = 0
End If

If Plan2.Range("M17") < 1 Then
Plan2.Range("M17") = 1
If ROT.Value = True Then
    Me.Hide
    Data2.Show
    Unload Me
Else
    TESTE = 1
    Unload Me
End If

Call Functions.SomaMomentoPerm(L / 2)
Call Functions.MomentoPermanenteBalanceada
End Sub

Private Sub Bt_RemArea_Click()
    elem = ListBox1.Text
    If elem = "" Then

```

```

        a = MsgBox("Selecione um elemento.",
vbOKOnly, "Atenção!")
        Exit Sub
    End If

    i = 2
    j = 0

    While i <= Plan2.Cells(Plan2.Rows.Count,
1).End(xlUp).Row And j < 1
        If Cdbl(Plan2.Cells(i, 1)) =
Cdbl(ListBox1.Text) Then
            Plan2.Range("A" & i, "I" & i).Delete
Shift:=xlUp
            j = 1
        End If
        i = i + 1
    Wend

'Atualiza ID
    For i = 2 To Ultima_B(1, 1)
        Plan2.Range("A" & i) = i - 1
    Next i
End Sub

Private Sub CheckBox1_Change()
    If CheckBox1 Then
        Image1.Visible = False
        ListBox1.Visible = False
        Bt_AddArea.Visible = False
        Bt_RemArea.Visible = False
        Bt_LimpArea.Visible = False
        Frame3.Visible = True
    Else
        Frame3.Visible = False
        Image1.Visible = True
        ListBox1.Visible = True
        Bt_AddArea.Visible = True
        Bt_RemArea.Visible = True
        Bt_LimpArea.Visible = True
    End If
End Sub

Private Sub ListBox1_DblClick(ByVal Cancel As
MSForms.ReturnBoolean)
    AdicionarAreaElementar.Show
End Sub

Private Sub TextBox1_Change()
    dado = TextBox1.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)

```

```

End If
If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox1.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
End If
End Sub

```

```

Private Sub TextBox10_Change()
If ComboBox1 = "" Then
Label30 = 1
End If
alphaE = Label30
dado = TextBox10.Text
x = 0
If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
        Me.TextBox10.Text = Mid(dado, 1,
Len(dado) - 1)
        x = 1
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
        Me.TextBox10.Text = Mid(dado, 1,
Len(dado) - 1)
        x = 1
    End If
    If ContaCarac(dado, ",") = 1 Then
        d = Split(dado, ",")
        If Len(d(1)) = 3 Then
            Me.TextBox10.Text = Mid(dado, 1,
Len(dado) - 1)
            x = 1
        End If
    End If
alphaI = 0.8 + 0.2 * TextBox10.Text / 80
If alphaI > 1 Then
alphaI = 1
End If
If x = 0 Then
    Eci = alphaE * 5600 * (TextBox10.Text ^ (1 / 2))
    Eci = Format(Eci, "0.00")
    Ecs = alphaI * Eci
    Ecs = Format(Ecs, "0.00")
    Label10 = Eci & " MPa"
    Label12 = Ecs & " MPa"
End If
Else
    Eci = 0
    Ecs = 0
    Label10 = ""
    Label12 = ""
End If
End Sub

```

```

Private Sub ComboBox1_Change()
If ComboBox1 = "basalto e diabásio" Then
Label30 = 1.2

```

```

End If
If ComboBox1 = "granito e gnaisse" Then
Label30 = 1
End If
If ComboBox1 = "calcário" Then
Label30 = 0.9
End If
If ComboBox1 = "arenito" Then
Label30 = 0.7
End If
End Sub

```

```

Private Sub TextBox11_Change()
dado = TextBox11.Text
If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
Then
        Me.TextBox11.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
End If
End Sub

```

```

Private Sub TextBox2_Change()
dado = TextBox2.Text
If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
        Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
        Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
        d = Split(dado, ",")
        If Len(d(1)) = 4 Then Me.TextBox2.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
End If
End Sub

```

```

Private Sub TextBox3_Change()
dado = TextBox3.Text
If Len(dado) = 4 Then
    Me.TextBox3.Text = Mid(dado, 1, Len(dado) - 1)
    Exit Sub
End If
If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
Then Me.TextBox3.Text = Mid(dado, 1, Len(dado) - 1)
    End If
End Sub

```

```

Private Sub TextBox4_Change()
dado = TextBox4.Text
If Len(dado) > 0 Then

```

```

    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
    Me.TextBox4.Text = Mid(dado, 1, Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
    Me.TextBox4.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox4.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox5_Change()
    dado = TextBox5.Text
    If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
    Me.TextBox5.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
    Me.TextBox5.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox5.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox6_Change()
    dado = TextBox6.Text
    If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
    Me.TextBox6.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
    Me.TextBox6.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox6.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox7_Change()
    dado = TextBox7.Text
    If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
    Me.TextBox7.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
    Me.TextBox7.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox7.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox8_Change()
    dado = TextBox8.Text
    If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
    Me.TextBox8.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
    Me.TextBox8.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox8.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox9_Change()
    dado = TextBox9.Text
    If Len(dado) > 0 Then
    If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
    Me.TextBox9.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
    Me.TextBox9.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
    d = Split(dado, ",")
    If Len(d(1)) = 4 Then Me.TextBox9.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

End If
End Sub

Private Sub UserForm_Initialize()
    L = Plan3.Range("B12")
    Atualiza_LB_Secao
    If SecaoConhecida = 1 Then CheckBox1.Value =
True
    Image1.Picture = Plan2.Image1.Picture
    TextBox1 = Plan3.Range("B6")
    TextBox2 = Plan3.Range("B12")
    TextBox3 = Plan3.Range("C13")
    TextBox4 = Plan3.Range("B16")
    TextBox10 = Plan3.Range("B17")
    TextBox5 = Plan3.Range("B7")
    TextBox6 = Plan3.Range("D8")
    TextBox7 = Plan3.Range("D9")
    TextBox8 = Plan3.Range("E10")
    TextBox9 = Plan3.Range("E11")
    TextBox11 = Plan3.Range("E20")
    ComboBox1.RowSource =
"BIBLIOTECA!CQ30:CQ33"
    Call Functions.SomaMomentoPerm(L / 2)
    Call Functions.MomentoPermanenteBalanceada
End Sub

Private Sub Data1_QueryClose(Cancel As Integer,
CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        Cancel = False
        Application.Quit
    Application.DisplayAlerts = False
    End If
End Sub

```

Adicionar área elementar

```

Private Sub Bt_Retang_BeforeUpdate(ByVal
Cancel As MSForms.ReturnBoolean)
    Bt_Triang = False
End Sub

Private Sub Bt_Triang_BeforeUpdate(ByVal
Cancel As MSForms.ReturnBoolean)
    Bt_Retang = False
End Sub

Private Sub CommandButton1_Click()
    ult = Ultima_B(1, 1) + 1
    If Bt_Retang = False And Bt_Triang = False
Then
        MsgBox "Selecione Seção RETANGULAR
ou TRIANGULAR", vbExclamation, "Seleciona
forma da seção"
        Exit Sub
    End If
    If tb_base = "" Or tb_alt = "" Or tb_y = "" Then
        MsgBox "Preencha todos os campos",
vbExclamation, "Dados em branco"

```

```

Exit Sub
End If
Plan2.Range("A" & ult) = ult - 1
Plan2.Range("B" & ult) = tb_base
Plan2.Range("C" & ult) = tb_alt
Plan2.Range("D" & ult) = tb_y
If Bt_Retang = True Then Plan2.Range("E" &
ult) = "Ret"
If Bt_Triang = True Then Plan2.Range("E" &
ult) = "Tri"
    Atualiza_LB_Secao
    Unload Me
End Sub

Private Sub tb_alt_Change()
    dado = tb_alt.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
            Me.tb_alt.Text = Mid(dado, 1, Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.tb_alt.Text = Mid(dado, 1, Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.tb_alt.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

Private Sub tb_base_Change()
    dado = tb_base.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
            Me.tb_base.Text = Mid(dado, 1, Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.tb_base.Text = Mid(dado, 1, Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.tb_base.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

Private Sub tb_y_Change()
    dado = tb_y.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
            Me.tb_y.Text = Mid(dado, 1, Len(dado) - 1)
        End If

```

```

    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
        Me.tb_y.Text = Mid(dado, 1, Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
        d = Split(dado, ",")
        If Len(d(1)) = 6 Then Me.tb_y.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
End If
End Sub

```

Data 2

```

Private TESTE As Integer 'Testa se o formulário foi
fechado pelo X
Private RESETA As Integer 'Testa se o nº de cabos
deve ser resetado (caso de se alterar Data2 após
Data3)

```

```

Private Sub btOK_2_Click()
    If Not OB_Pre And Not OB_Pos Then
        MsgBox "Escolha o Tipo de Protensão.",
vbOKOnly, "Dados insuficientes"
        Exit Sub
    Else
        If OB_Pre Then
            Plan3.Range("C23") = OB_Pre.Caption
        ElseIf OB_Pos Then
            Plan3.Range("C23") = OB_Pos.Caption
        End If
    End If
    If Label2 = "" Then
        MsgBox "Informe o Nível de Protensão.",
vbOKOnly, "Dados insuficientes"
        Exit Sub
    Else
        If Plan3.Range("C24") <> Label2 Then
RESETA = 1
            Plan3.Range("C24") = Label2
            NivelProt = Plan3.Range("C24")
        End If
        If TextBox7 = "" Then
            MsgBox "Informe a Excentricidade.",
vbOKOnly, "Dados insuficientes"
            Exit Sub
        Else
            If TextBox7 <> Plan3.Range("B28") Then
RESETA = 1
                Plan3.Range("B28") = Cdbl(TextBox7)
            End If
            If PerdaEstimada = "" Then
                MsgBox "Informe a Perda de Carga
Estimada.", vbOKOnly, "Dados insuficientes"
                Exit Sub
            Else
                If PerdaEstimada <> Plan3.Range("B30")
Then RESETA = 1
                    Plan3.Range("B30") = Cdbl(PerdaEstimada)

```

```

        End If
        If (Not (OptionButton9 Or OptionButton10))
Then
            MsgBox "Informe a Tensão no ELS-F",
vbOKOnly, "Dados insuficientes"
            Exit Sub
        Else
            If OptionButton9 Then
                If Plan2.Range("O17") <> 1.5 Then
RESETA = 1
                    Plan2.Range("O17") = 1.5
                End If
            If OptionButton10 Then
                If Plan2.Range("O17") <> 1.2 Then
RESETA = 1
                    Plan2.Range("O17") = 1.2
                End If
            End If
            Plan3.Range("A22:K24").Copy
Destination:=Plan1.Range("A22")
            Estima_P
            Plan3.Range("D28:K31").Copy
Destination:=Plan1.Range("D28")
            If Plan3.Range("C35") >= 1 And RESETA = 1
Then
                Plan3.Range("C27") = ""
                Plan3.Range("B33") = 0
                Plan3.Range("B34") = 0
                Plan3.Range("C35") = 0
                Plan3.Range("B36") = 0
                Plan3.Range("B37") = 0
                Plan3.Range("A26:K38").Copy
Destination:=Plan1.Range("A26")
                ultC = Ultima_B(62, 62)
                If ultC > 10 Then
                    For i = 11 To ultC
                        Plan2.Range("BJ" & i) = ""
                        Plan2.Range("BK" & i) = ""
                    Next i
                End If
                MsgBox "Número de cabos e Força de
Protensão deverão ser recalculados.", vbOKOnly
                Plan2.Range("M17") = 2
            End If
            If Plan2.Range("M17") < 2 Then
                Plan2.Range("M17") = 2
            If Data1.ROT.Value = True Then
                Me.Hide
                Data3.Show
                Unload Me
            Else
                TESTE = 1
                Unload Me
            End If
        End Sub

Private Sub CheckBox1_Change()
    If CheckBox1 Then
        Frame24.Visible = True
        Frame3.Enabled = False

```

```

OB_I = False
OB_II = False
OB_III = False
OB_IV = False
Else
    Frame24.Visible = False
    Frame29.Visible = False
    Frame3.Enabled = True
End If
End Sub

Private Sub CommandButton1_Click()
    If Plan3.Range("B12") = "" Then
        MsgBox "Primeiro entre com o vão da viga"
        Exit Sub
    End If
    L = Plan3.Range("B12")
    ult = Ultima_B(18, 18) + 1
    If OptionButton4 = False And OptionButton5 =
False Then
        MsgBox "Selecione PERMANENTE ou
VARIÁVEL.", vbExclamation, "Dados
incompletos"
        Exit Sub
    End If
    If OptionButton1 = False And OptionButton2 =
False Then
        MsgBox "Selecione a carga a ser adicionada.",
vbExclamation, "Dados incompletos"
        Exit Sub
    End If
    If OptionButton1 Then
        If TextBox1 = "" Or TextBox2 = "" Then
            MsgBox "Preencha todos os campos.",
vbExclamation, "Dados incompletos"
            Exit Sub
        Else
            If CDbl(TextBox2.Text) > L Then
                MsgBox "Posição da carga não pode ser
maior que o vão da viga.", vbExclamation, "Erro"
                Exit Sub
            End If
            Plan2.Range("S" & ult) = "Conc."
            Plan2.Range("T" & ult) = TextBox1
            Plan2.Range("U" & ult) = TextBox2
            Plan2.Range("V" & ult) = "-"
        End If
    End If
    If OptionButton2 Then
        If TextBox1 = "" Or TextBox2 = "" Or
TextBox3 = "" Then
            MsgBox "Preencha todos os campos.",
vbExclamation, "Dados incompletos"
            Exit Sub
        Else
            If CDbl(TextBox2.Text) > L Or
CDbl(TextBox3.Text) > L Then
                MsgBox "Posição da carga não pode ser
maior que o vão da viga.", vbExclamation, "Erro"
                Exit Sub
            End If
            If CDbl(TextBox2) >= CDbl(TextBox3)
Then
                MsgBox "Posição final deve ser maior
que a posição inicial.", vbExclamation, "Erro"
                Exit Sub
            End If
            Plan2.Range("S" & ult) = "Dist."
            Plan2.Range("T" & ult) = TextBox1
            Plan2.Range("U" & ult) = TextBox2
            Plan2.Range("V" & ult) = TextBox3
        End If
    End If
    If OptionButton4 Then
        Plan2.Range("W" & ult) = "Perm"
        Plan2.Range("X" & ult) = "-"
        Plan2.Range("Y" & ult) = "-"
        Plan2.Range("Z" & ult) = "-"
    Else
        If TextBox4 = "" Or TextBox5 = "" Or
TextBox6 = "" Then
            MsgBox "Preencha os valores de Psi.",
vbExclamation, "Dados incompletos"
            Exit Sub
        End If
        Plan2.Range("W" & ult) = "Var"
        Plan2.Range("X" & ult) = TextBox4
        Plan2.Range("Y" & ult) = TextBox5
        Plan2.Range("Z" & ult) = TextBox6
    End If
    Plan2.Range("R" & ult) = ult - 1
    Atualiza_LB_Carga
    OptionButton1 = False
    OptionButton2 = False
    OptionButton4 = False
    OptionButton5 = False
    OptionButton6 = False
    TextBox1 = ""
    TextBox2 = ""
    TextBox3 = ""
    TextBox4 = ""
    TextBox5 = ""
    TextBox6 = ""
    Label3.Visible = False
    Label4.Visible = False
    Label5.Visible = False
    Label6.Visible = False
    Label7.Visible = False
    Label8.Visible = False
    Label9.Visible = False
    Label10.Visible = False
    Label11.Visible = False
    Label12.Visible = False
    TextBox1.Visible = False
    TextBox2.Visible = False
    TextBox3.Visible = False
    TextBox4.Visible = False
    TextBox5.Visible = False
    TextBox6.Visible = False
    CommandButton4.Visible = False

```

```

RESETA = 1
Call Functions.SomaMomentoPerm(L / 2)
Call Functions.MomentoPermanenteBalanceada
End Sub

```

```

Private Sub CommandButton2_Click()
    L = Plan3.Range("B12")
    elem = ListBox1.Text
    If elem = "" Then
        a = MsgBox("Selecione um elemento.",
vbOKOnly, "Atenção!")
        Exit Sub
    End If

```

```

i = 2
j = 0
While i <= Plan2.Cells(Plan2.Rows.Count,
18).End(xlUp).Row And j < 1
    If CDb(Plan2.Cells(i, 18)) =
CDbl(ListBox1.Text) Then
        Plan2.Range("R" & i, "AC" & i).Delete
Shift:=xlUp
        j = 1
    End If
    i = i + 1
Wend

```

```

'Atualiza ID
For i = 2 To Ultima_B(18, 18)
    Plan2.Range("R" & i) = i - 1
Next i
RESETA = 1
Call Functions.SomaMomentoPerm(L / 2)
Call Functions.MomentoPermanenteBalanceada
End Sub

```

```

Private Sub CommandButton3_Click()
    Data2.Width = 600
End Sub

```

```

Private Sub CommandButton4_Click()
    Data2.Width = 900
End Sub

```

```

Private Sub Label63_Click()
End Sub

```

```

Private Sub OB_I_Click()
    If (OB_I And OB_Pre) Or (OB_I And OB_Pos)
Then Label2 = "Nível 1 - Protensão Parcial"
    If (OB_I And OB_Pre) Or (OB_I And OB_Pos)
Then Frame29.Visible = True
End Sub

```

```

Private Sub OB_II_Click()
    If (OB_II And OB_Pos) Then Label2 = "Nível 1
- Protensão Parcial"
    If (OB_II And OB_Pos) Then Frame29.Visible =
True

```

```

    If (OB_II And OB_Pre) Then Label2 = "Nível 2
- Protensão Limitada"
    If (OB_II And OB_Pre) Then Frame29.Visible =
False
End Sub

```

```

Private Sub OB_III_Click()
    If (OB_III And OB_Pos) Then Label2 = "Nível 2
- Protensão Limitada"
    If (OB_III And OB_Pos) Then Frame29.Visible
= False
    If (OB_III And OB_Pre) Then Label2 = "Nível 3
- Protensão Completa"
    If (OB_III And OB_Pre) Then Frame29.Visible
= False
End Sub

```

```

Private Sub OB_IV_Click()
    If (OB_IV And OB_Pos) Then Label2 = "Nível 2
- Protensão Limitada"
    If (OB_IV And OB_Pos) Then Frame29.Visible
= False
    If (OB_IV And OB_Pre) Then Label2 = "Nível 3
- Protensão Completa"
    If (OB_IV And OB_Pre) Then Frame29.Visible
= False
End Sub

```

```

Private Sub OB_Pos_Click()
    If (OB_I And OB_Pos) Or (OB_II And OB_Pos)
Then Label2 = "Nível 1 - Protensão Parcial"
    If (OB_I And OB_Pos) Or (OB_II And OB_Pos)
Then Frame29.Visible = True
    If (OB_III And OB_Pos) Or (OB_IV And
OB_Pos) Then Label2 = "Nível 2 - Protensão
Limitada"
    If (OB_III And OB_Pos) Or (OB_IV And
OB_Pos) Then Frame29.Visible = False
End Sub

```

```

Private Sub OB_Pre_Click()
    If (OB_I And OB_Pre) Then Label2 = "Nível 1 -
Protensão Parcial"
    If (OB_I And OB_Pre) Then Frame29.Visible =
True
    If (OB_II And OB_Pre) Then Label2 = "Nível 2
- Protensão Limitada"
    If (OB_II And OB_Pre) Then Frame29.Visible =
False
    If (OB_III And OB_Pre) Or (OB_IV And
OB_Pre) Then Label2 = "Nível 3 - Protensão
Completa"
    If (OB_III And OB_Pre) Or (OB_IV And
OB_Pre) Then Frame29.Visible = False
End Sub

```

```

Private Sub OptionButton1_Click()
    Label4.Visible = False
    Label10.Visible = False
    TextBox3.Visible = False

```



```

Label3 = "Valor"
Label11 = "Posição"
Label8 = "KN"
Label11.Visible = True
Label3.Visible = True
Label8.Visible = True
Label9.Visible = True
TextBox1.Visible = True
TextBox2.Visible = True
End Sub

Private Sub OptionButton11_Change()
    If OptionButton11 Then
        TextBox8.Visible = True
        Label62.Visible = True
    End If
    If Not OptionButton11 Then
        TextBox8.Visible = False
        Label62.Visible = False
    End If
End Sub

Private Sub OptionButton2_Click()
    Label3 = "Valor"
    Label11 = "Posição Inicial"
    Label4 = "Posição Final"
    Label8 = "KN/m"
    Label11.Visible = True
    Label3.Visible = True
    Label4.Visible = True
    Label8.Visible = True
    Label9.Visible = True
    Label10.Visible = True
    TextBox1.Visible = True
    TextBox2.Visible = True
    TextBox3.Visible = True
End Sub

Private Sub OptionButton4_Click()
    Label5.Visible = False
    Label6.Visible = False
    Label7.Visible = False
    TextBox4.Visible = False
    TextBox5.Visible = False
    TextBox6.Visible = False
    CommandButton4.Visible = False
    Label12.Visible = False
End Sub

Private Sub OptionButton5_Click()
    Label5.Visible = True
    Label6.Visible = True
    Label7.Visible = True
    TextBox4.Visible = True
    TextBox5.Visible = True
    TextBox6.Visible = True
    CommandButton4.Visible = True
    Label12.Visible = True
End Sub

Private Sub OptionButton6_Click()
    Label2 = "Nível 1 - Protensão Parcial"
    Frame29.Visible = True
End Sub

Private Sub OptionButton7_Click()
    Label2 = "Nível 2 - Protensão Limitada"
    Frame29.Visible = False
End Sub

Private Sub OptionButton8_Click()
    Label2 = "Nível 3 - Protensão Completa"
    Frame29.Visible = False
End Sub

Private Sub PerdaEstimada_Change()
    dado = PerdaEstimada.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.PerdaEstimada.Text = Mid(dado, 1,
            Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
        Len(dado), 1) = "," Then
            Me.PerdaEstimada.Text = Mid(dado, 1,
            Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then
                Me.PerdaEstimada.Text = Mid(dado, 1, Len(dado) - 1)
            End If
        End If
    End Sub

Private Sub TextBox1_Change()
    dado = TextBox1.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox1.Text = Mid(dado, 1,
            Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
        Len(dado), 1) = "," Then
            Me.TextBox1.Text = Mid(dado, 1,
            Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.TextBox1.Text =
            Mid(dado, 1, Len(dado) - 1)
        End If
    End Sub

Private Sub TextBox2_Change()
    dado = TextBox2.Text
    If Len(dado) > 0 Then

```

```

    If Not IsNumeric(Mid(dado, Len(dado), 1))
    And Mid(dado, Len(dado), 1) <> "," Then
        Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
        Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
    End If
    If ContaCarac(dado, ",") = 1 Then
        d = Split(dado, ",")
        If Len(d(1)) = 4 Then Me.TextBox2.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox3_Change()
    dado = TextBox3.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
    And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.TextBox3.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub TextBox4_Change()
    dado = TextBox4.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
    And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox4.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox4.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 2 Then Me.TextBox4.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub TextBox5_Change()
    dado = TextBox5.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
    And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox5.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox5.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 2 Then Me.TextBox5.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub TextBox6_Change()
    dado = TextBox6.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
    And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox6.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox6.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 2 Then Me.TextBox6.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub TextBox8_Change()
    dado = TextBox8.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
    And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox8.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox8.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.TextBox8.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

```

End If
Plan6.Range("B6") = TextBox8.Text
Call Functions.MomentoPermanenteBalanceada
End Sub

Private Sub TextBox7_Change()
    dado = TextBox7.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox7.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox7.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.TextBox7.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

Private Sub UserForm_Activate()
If Plan2.Range("M17") = 0 Then
    MsgBox "Abrir primeiro janelas anteriores!"
    Unload Data2
    Data2.Hide
End If
End Sub

Private Sub UserForm_Initialize()
    Data2.Width = 600
    Atualiza_LB_Carga
    Label2 = " "
    Select Case Plan3.Range("C23")
        Case "Pré-Tração"
            OB_Pre = True
        Case "Pós-Tração"
            OB_Pos = True
    End Select
    Label58 = Plan3.Range("D8") & " m"
    TextBox7 = Plan3.Range("B28")
    fator = Plan2.Range("O17")
    PerdaEstimada = Plan3.Range("B30")
    If fator = 1.5 Then
        OptionButton9 = True
    ElseIf fator = 1.2 Then
        OptionButton10 = True
    End If
    Plan3.Range("D28:K31").Interior.Color =
RGB(255, 255, 255)
    Plan3.Range("D28:K31").Clear
    RESETA = 0
End Sub

```

```

Private Sub Data2_QueryClose(Cancel As Integer,
CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        Cancel = False
        Application.Quit
    Application.DisplayAlerts = False
    End If
End Sub

```

Data 3

```

Private TESTE As Integer 'Testa se o formulário foi
fechado pelo X
Private DATA3ABERTO As Integer

Private Sub btOK_3_Click()
    Plan3.Range("B33") = Sigma_pi
    Plan3.Range("B34") =
Application.WorksheetFunction.RoundUp(Ap_est, 3)
    Plan3.Range("C35") = CDbI(TextBox1)
    Plan3.Range("B36") =
Application.WorksheetFunction.RoundUp(Ap_ef, 3)
    Plan3.Range("B37") = CDbI(Pi_ef)
    ult = Ultima_B(62, 62)
    Plan3.Range("C38") = ult - 10
    If RESETACABO = 1 Then
        Estima_Cabos 'preenche lista de cabos e
excentricidades estimadas
        Tracado_Cabos
        Cabo_Equivalente
    End If
    DATA3ABERTO = 0
    RESETACABO = 0
    Plan3.Range("A26:K38").Copy
Destination:=Plan1.Range("A26")
    If Plan2.Range("M17") < 3 Then
        Plan2.Range("M17") = 3
        If Data1.ROT.Value = True Then
            Me.Hide
            Data4.Show
            Unload Me
        Else
            TESTE = 1
            Unload Me
        End If
    End Sub

Private Sub CommandButton1_Click()
    AddAco.Show
End Sub

Private Sub CommandButton2_Click()
    elem = ListBox1.Text
    If elem = "" Then
        a = MsgBox("Selecione um elemento.",
vbOKOnly, "Atenção!")
    Exit Sub
    End If
    i = 2

```

```

j = 0
If OptionButton1 Then
    While i <= Plan2.Cells(Rows.Count,
31).End(xlUp).Row And j < 1
        If CDb(Plan2.Cells(i, 31)) =
CDbl(ListBox1.Text) Then
            Plan2.Range("AE" & i, "AL" & i).Delete
Shift:=xlUp
            j = 1
            End If
            i = i + 1
        Wend
        'Atualiza ID
        For i = 2 To Ultima_B(31, 31)
            Plan2.Range("AE" & i) = i - 1
        Next i
    End If
    If OptionButton2 Then
        While i <= Plan2.Cells(Rows.Count,
42).End(xlUp).Row And j < 1
            If CDb(Plan2.Cells(i, 42)) =
CDbl(ListBox1.Text) Then
                Plan2.Range("AP" & i, "AW" & i).Delete
Shift:=xlUp
                j = 1
                End If
                i = i + 1
            Wend
            'Atualiza ID
            For i = 2 To Ultima_B(42, 42)
                Plan2.Range("AP" & i) = i - 1
            Next i
        End If
        If OptionButton3 Then
            While i <= Plan2.Cells(Rows.Count,
53).End(xlUp).Row And j < 1
                If CDb(Plan2.Cells(i, 53)) =
CDbl(ListBox1.Text) Then
                    Plan2.Range("BA" & i, "BH" & i).Delete
Shift:=xlUp
                    j = 1
                    End If
                    i = i + 1
                Wend
                'Atualiza ID
                For i = 2 To Ultima_B(53, 53)
                    Plan2.Range("BA" & i) = i - 1
                Next i
            End If
            Plan3.Range("B32") = ""
            Label3 = ""
            Plan2.Range("BK2") = ""
            Plan2.Range("BK3") = ""
        End Sub

Private Sub CommandButton3_Click()
    'Selecciona material
    elem = ListBox1.Text
    i = 2
    j = 0

    If OptionButton1 Then
        While i <= Plan2.Cells(Rows.Count,
31).End(xlUp).Row And j < 1
            If CDb(Plan2.Cells(i, 31)) =
CDbl(ListBox1.Text) Then
                Plan2.Range("BK2") = "RN"
                Plan2.Range("BK3") =
Plan2.Range("AE" & i)
                Plan2.Range("BK4") =
Plan2.Range("AK" & i)
                Plan2.Range("BK5") = Plan2.Range("AI" & i)
                Plan2.Range("BK6") = Plan2.Range("AL" & i)
                Plan2.Range("BK7") =
Plan2.Range("AM" & i)
                Label3 = Plan2.Range("AF" & i)
                Plan3.Range("B32") = Label3
                j = 1
            End If
            i = i + 1
        Wend
    End If
    If OptionButton2 Then
        While i <= Plan2.Cells(Rows.Count,
42).End(xlUp).Row And j < 1
            If CDb(Plan2.Cells(i, 42)) =
CDbl(ListBox1.Text) Then
                Plan2.Range("BK2") = "RB"
                Plan2.Range("BK3") =
Plan2.Range("AP" & i)
                Plan2.Range("BK4") =
Plan2.Range("AV" & i)
                Plan2.Range("BK5") =
Plan2.Range("AS" & i)
                Plan2.Range("BK6") =
Plan2.Range("AW" & i)
                Plan2.Range("BK7") =
Plan2.Range("AX" & i)
                Label3 = Plan2.Range("AQ" & i)
                Plan3.Range("B32") = Label3
                j = 1
            End If
            i = i + 1
        Wend
    End If
    If OptionButton3 Then
        While i <= Plan2.Cells(Rows.Count,
53).End(xlUp).Row And j < 1
            If CDb(Plan2.Cells(i, 53)) =
CDbl(ListBox1.Text) Then
                Plan2.Range("BK2") = "Barra"
                Plan2.Range("BK3") =
Plan2.Range("BA" & i)
                Plan2.Range("BK4") =
Plan2.Range("BG" & i)
                Plan2.Range("BK5") =
Plan2.Range("BD" & i)
                Plan2.Range("BK6") =
Plan2.Range("BH" & i)
                Plan2.Range("BK7") = Plan2.Range("BI" & i)
                Label3 = "-" & Plan2.Range("BB" & i)
            End If
        End If
    End If
End Sub

```

```

        Plan3.Range("B32") = Label3
        j = 1
    End If
    i = i + 1
Wend
End If
SelecCLASSE = Plan2.Range("BK2")
SelecID = Plan2.Range("BK3")
selecFPTK = Plan2.Range("BK4")
selecAREA = Plan2.Range("BK5") / 100
selecFPYK = Plan2.Range("BK6")
'Calcula Sigma_pi
If Plan3.Range("C23") = "Pré-Tração" Then
    If SelecCLASSE = "RB" Then
        Sigma_pi1 = 0.77 * selecFPTK
        Sigma_pi2 = 0.85 * selecFPYK
        If Sigma_pi1 < Sigma_pi2 Then
            Sigma_pi = Sigma_pi1
        Else
            Sigma_pi = Sigma_pi2
        End If
    End If
    If SelecCLASSE = "RN" Then
        Sigma_pi1 = 0.77 * selecFPTK
        Sigma_pi2 = 0.9 * selecFPYK
        If Sigma_pi1 < Sigma_pi2 Then
            Sigma_pi = Sigma_pi1
        Else
            Sigma_pi = Sigma_pi2
        End If
    End If
End If
If Plan3.Range("C23") = "Pós-Tração" Then
    If SelecCLASSE = "RB" Then
        Sigma_pi1 = 0.74 * selecFPTK
        Sigma_pi2 = 0.82 * selecFPYK
        If Sigma_pi1 < Sigma_pi2 Then
            Sigma_pi = Sigma_pi1
        Else
            Sigma_pi = Sigma_pi2
        End If
    End If
    If SelecCLASSE = "RN" Then
        Sigma_pi1 = 0.74 * selecFPTK
        Sigma_pi2 = 0.87 * selecFPYK
        If Sigma_pi1 < Sigma_pi2 Then
            Sigma_pi = Sigma_pi1
        Else
            Sigma_pi = Sigma_pi2
        End If
    End If
    If SelecCLASSE = "Barra" Then
        Sigma_pi1 = 0.72 * selecFPTK
        Sigma_pi2 = 0.88 * selecFPYK
        If Sigma_pi1 < Sigma_pi2 Then
            Sigma_pi = Sigma_pi1
        Else
            Sigma_pi = Sigma_pi2
        End If
    End If
End If

```

```

    End If
'Calcula Ap,est e Numero de cabos
    Pi_est = Plan3.Range("B31")

    Ap_est = Pi_est / (Sigma_pi / 10)
    NumCabos = Ap_est / selecAREA
    Label5 = Sigma_pi & "MPa"
    Label7 = FormatNumber(Ap_est, 2) & " cm²"
    TextBox1 = 0
    TextBox1 =
ArredondarParaCima(CDbl(NumCabos))
    Plan3.Range("B33") = Sigma_pi
    Plan3.Range("B34") =
Application.WorksheetFunction.RoundUp(Ap_est, 3)
    Plan3.Range("C35") = CDbl(TextBox1)
    Plan3.Range("B36") =
Application.WorksheetFunction.RoundUp(Ap_ef, 3)
    Plan3.Range("B37") = CDbl(Pi_cf)
'CABOS
    Plan2.Range("BJ11") = 1
    Plan2.Range("BK11") = CDbl(TextBox1)
    ultC = Ultima_B(62, 62)
    If ultC > 11 Then
        For i = 12 To ultC
            Plan2.Range("BJ" & i) = ""
            Plan2.Range("BK" & i) = ""
        Next i
    End If
'If Plan3.Range("C24") <> "Nível 1 - Protensão
Parcial" Then
    Me.Repaint
    DesenhaViga 50, 480, 750, 1, 150
    DesenhaFuso 50, 480, 750, 1
'End If
End Sub

Private Sub CommandButton4_Click()
    RESETACABO = 1
    ult = Ultima_B(62, 62)

    If TextBox1 = "" Or TextBox1 = 0 Then
        MsgBox "Antes calcule o número de fios ou
cordoalhas.", vbExclamation, "Dados incompletos"
        Exit Sub
    End If
    Plan2.Range("BJ" & ult + 1) = ult - 9
    Ncabos = Plan2.Range("BJ" & ult + 1)
    Nfios = TextBox1

    base =
Application.WorksheetFunction.RoundDown((Nfios
/ Ncabos), 0)
    sobra = Nfios - (base * Ncabos)
    For i = 1 To sobra
        Plan2.Range("BK" & ult + 1 - Ncabos + i) =
base + 1
    Next i
    For j = sobra + 1 To Ncabos
        Plan2.Range("BK" & ult + 1 - Ncabos + j) =
base
    End For

```

```

Next j
Atualiza_LB_Cabos
End Sub

Private Sub CommandButton5_Click()
    RESETACABO = 1
    ult = Ultima_B(62, 62)
    If TextBox1 = "" Or TextBox1 = 0 Then
        MsgBox "Antes calcule o número de fios ou cordoalhas.", vbExclamation, "Dados incompletos"
        Exit Sub
    End If
    If ult <= 11 Then
        MsgBox "O número mínimo de cabos deve ser 1.", vbExclamation, "Função indisponível"
        Exit Sub
    End If
    Plan2.Range("BJ" & ult) = ""
    Plan2.Range("BK" & ult) = ""
    Ncabos = Plan2.Range("BJ" & ult - 1)
    Nfios = TextBox1
    base =
Application.WorksheetFunction.RoundDown((Nfios / Ncabos), 0)
    sobra = Nfios - (base * Ncabos)
    For i = 1 To sobra
        Plan2.Range("BK" & ult - 1 - Ncabos + i) =
base + 1
    Next i
    For j = sobra + 1 To Ncabos
        Plan2.Range("BK" & ult - 1 - Ncabos + j) = base
    Next j
    Atualiza_LB_Cabos
End Sub

Private Sub CommandButton7_Click()
    elem = ListBox2.Text
    If elem = "" Then
        a = MsgBox("Selecione um elemento.", vbOKOnly, "Atenção!")
        Exit Sub
    End If
    Plan2.Range("BK9") = elem
    EditaCabo.Show
End Sub

Private Sub ListBox1_Change()
    elem = ListBox1.Text
    If elem = "" Then
        CommandButton2.Enabled = False
        CommandButton3.Enabled = False
    Else
        CommandButton3.Enabled = True
    End If
    If OptionButton1 Then
        If elem <= 4 Then
            CommandButton2.Enabled = False
        Else
            CommandButton2.Enabled = True
        End If
    End If

End If
If OptionButton2 Then
    If elem <= 17 Then
        CommandButton2.Enabled = False
    Else
        CommandButton2.Enabled = True
    End If
End If
If OptionButton3 Then
    If elem = 1 Then
        CommandButton2.Enabled = False
    Else
        CommandButton2.Enabled = True
    End If
End If
End Sub

Private Sub OptionButton1_Change()
    If OptionButton1 Then
        ult = Ultima_B(31, 31)
        Data3.ListBox1.ColumnCount = 8
        Data3.ListBox1.RowSource =
"BIBLIOTECA!AE2:AL" & ult
        CommandButton1.Enabled = True
    End If
End Sub

Private Sub OptionButton2_Change()
    If OptionButton2 Then
        ult = Ultima_B(42, 42)
        Data3.ListBox1.ColumnCount = 8
        Data3.ListBox1.RowSource =
"BIBLIOTECA!AP2:AW" & ult
        CommandButton1.Enabled = True
    End If
End Sub

Private Sub OptionButton3_Change()
    If OptionButton3 Then
        ult = Ultima_B(53, 53)
        Data3.ListBox1.ColumnCount = 8
        Data3.ListBox1.RowSource =
"BIBLIOTECA!BA2:BH" & ult
        CommandButton1.Enabled = True
    End If
End Sub

Private Sub TextBox1_Change()
    RESETACABO = 1
    dado = TextBox1.Text
    If Len(dado) = 4 Then
        Me.TextBox1.Text = Mid(dado, 1, Len(dado) - 1)
        Exit Sub
    End If
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
        Exit Sub
    End Sub

```

```

End If
selecAREA = Plan2.Range("BK5") / 100
Ap_ef = TextBox1 * selecAREA
Pi_ef = Ap_ef * (Sigma_pi / 10)
Pi_ef = Format(Pi_ef, "0.00")
Label12 = Ap_ef & " cm²"
Label9 = Pi_ef & " KN"
Else
Label9 = 0
Label12 = 0
End If
Plan3.Range("C35") = TextBox1
Plan3.Range("B36") = Ap_ef
Plan3.Range("B37") = Pi_ef
'If Plan3.Range("C24") <> "Nível 1 - Protensão
Parcial" Then
Me.Repaint
DesenhaViga 50, 480, 750, 1, 150
DesenhaFuso 50, 480, 750, 1
'End If
End Sub

Private Sub TextBox2_Change()
dado = TextBox2.Text
If Len(dado) > 0 Then
If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") = 1 Then
d = Split(dado, ",")
If Len(d(1)) = 4 Then Me.TextBox2.Text =
Mid(dado, 1, Len(dado) - 1)
End If
End If
If TextBox2 > 0 And TextBox2 <> "" And
Plan3.Range("B37") > 0 Then
'If Plan3.Range("C24") <> "Nível 1 -
Protensão Parcial" Then
Me.Repaint
DesenhaViga 50, 480, 750, TextBox2, 150
DesenhaFuso 50, 480, 750, TextBox2
'End If
TextBox2 = esc
End If
End Sub

Private Sub UserForm_Activate()
'If Plan3.Range("C24") <> "Nível 1 - Protensão
Parcial" Then
Me.Repaint
DesenhaViga 50, 480, 750, 1, 150
'End If
TextBox2 = esc

```

```

TextBox1 = Plan3.Range("C35")
Atualiza_LB_Cabos
RESETACABO = 0
If Plan2.Range("M17") < 2 Then
MsgBox "Abrir primeiro janelas anteriores!"
Unload Data3
Data3.Hide
End If
End Sub

```

```

Private Sub UserForm_Initialize()
'If Plan3.Range("C24") = "Nível 1 - Protensão
Parcial" Then
'Label17.Visible = False
'Label18.Visible = False
'Label19.Visible = False
'Label20.Visible = False
'TextBox2.Visible = False
'End If
If Not (OptionButton1 Or OptionButton2 Or
OptionButton3) Then
CommandButton1.Enabled = False
CommandButton2.Enabled = False
CommandButton3.Enabled = False
End If
If Plan3.Range("C23") = "Pré-Tração" Then
OptionButton3.Enabled = False
Classe = Plan2.Range("BK2")
If Classe = "RN" Then OptionButton1 = True
If Classe = "RB" Then OptionButton2 = True
If Classe = "Barra" Then OptionButton3 = True
Label3 = Plan3.Range("B32")
Label5 = Plan3.Range("B33") & " MPa"
Sigma_pi = Plan3.Range("B33")
Label23 = Plan3.Range("B31") & " KN"
Label7 = Plan3.Range("B34") & " cm²"
Ap_est = Plan3.Range("B34")
Label12 = Plan3.Range("B36") & " cm²"
Ap_ef = Plan3.Range("B36")
Label9 = Plan3.Range("B37") & " KN"
Pi_ef = Plan3.Range("B37")
End Sub

```

```

Private Sub Data3_QueryClose(Cancel As Integer,
CloseMode As Integer)
If CloseMode = vbFormControlMenu Then
Cancel = False
Application.Quit
Application.DisplayAlerts = False
End If
End Sub

```

Adicionar aço

```

Private Sub CommandButton1_Click()
If Data3.OptionButton1 Then
ult = Ultima_B(31, 31)
prox = ult + 1
Plan2.Range("AE" & prox) = ult

```

```

Plan2.Range("AF" & prox) = TextBox1
Plan2.Range("AG" & prox) = TextBox2
Plan2.Range("AH" & prox) = TextBox3
Plan2.Range("AI" & prox) = TextBox4
Plan2.Range("AJ" & prox) = TextBox5
Plan2.Range("AK" & prox) = TextBox6
Plan2.Range("AL" & prox) = TextBox7
'Atualiza ListBox
Data3.ListBox1.RowSource =
"BIBLIOTECA!AE2:AL" & prox
End If
If Data3.OptionButton2 Then
ult = Ultima_B(42, 42)
prox = ult + 1
Plan2.Range("AP" & prox) = ult
Plan2.Range("AQ" & prox) = TextBox1
Plan2.Range("AR" & prox) = TextBox2
Plan2.Range("AS" & prox) = TextBox3
Plan2.Range("AT" & prox) = TextBox4
Plan2.Range("AU" & prox) = TextBox5
Plan2.Range("AV" & prox) = TextBox6
Plan2.Range("AW" & prox) = TextBox7
Data3.ListBox1.RowSource =
"BIBLIOTECA!AP2:AW" & prox
End If
If Data3.OptionButton3 Then
ult = Ultima_B(53, 53)
prox = ult + 1
Plan2.Range("BA" & prox) = ult
Plan2.Range("BB" & prox) = TextBox1
Plan2.Range("BC" & prox) = TextBox2
Plan2.Range("BD" & prox) = TextBox3
Plan2.Range("BE" & prox) = TextBox4
Plan2.Range("BF" & prox) = TextBox5
Plan2.Range("BG" & prox) = TextBox6
Plan2.Range("BH" & prox) = TextBox7
Data3.ListBox1.RowSource =
"BIBLIOTECA!BA2:BH" & prox
End If
Unload Me
End Sub

Private Sub TextBox2_Change()
dado = TextBox2
If Len(dado) > 0 Then
If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") = 1 Then
d = Split(dado, ",")
If Len(d(1)) = 4 Then Me.TextBox2.Text =
Mid(dado, 1, Len(dado) - 1)
End If
End If

```

```

End If
End Sub

Private Sub TextBox3_Change()
dado = TextBox3
If Len(dado) > 0 Then
If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") = 1 Then
d = Split(dado, ",")
If Len(d(1)) = 4 Then Me.TextBox3.Text =
Mid(dado, 1, Len(dado) - 1)
End If
End If
End Sub

Private Sub TextBox4_Change()
dado = TextBox4
If Len(dado) > 0 Then
If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
Me.TextBox4.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
Me.TextBox4.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") = 1 Then
d = Split(dado, ",")
If Len(d(1)) = 4 Then Me.TextBox4.Text =
Mid(dado, 1, Len(dado) - 1)
End If
End If
End Sub

Private Sub TextBox5_Change()
dado = TextBox5
If Len(dado) > 0 Then
If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
Me.TextBox5.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
Me.TextBox5.Text = Mid(dado, 1,
Len(dado) - 1)
End If
If ContaCarac(dado, ",") = 1 Then
d = Split(dado, ",")

```



```

    If Len(d(1)) = 4 Then Me.TextBox5.Text =
Mid(dado, 1, Len(dado) - 1)
    End If
    End If
End Sub

```

```

Private Sub TextBox6_Change()
    dado = TextBox6
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
Then
            Me.TextBox6.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub TextBox7_Change()
    dado = TextBox7
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
Then
            Me.TextBox7.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub UserForm_Initialize()
Label2.Caption = ChrW(632) & " Nominal"
End Sub

```

Editar Cabo

```

Private Sub CommandButton1_Click()
    RESETACABO = 1
    ult = Ultima_B(62, 62)
    c = Plan2.Range("BK9")
    soma = 0
    Plan2.Range("BK" & (10 + c)) =
CDBl(TextBox1)
    For i = 11 To ult
        soma = soma + Plan2.Range("BK" & i)
    Next i
    Data3.TextBox1 = soma
    selecAREA = Plan2.Range("BK5") / 100
    Ap_ef = Data3.TextBox1 * selecAREA
    Pi_ef = Ap_ef * (Sigma_pi / 10)
    Pi_ef = Format(Pi_ef, "0.00")
    Label12 = Ap_ef & " cm²"
    Label9 = Pi_ef & " KN"
    Plan3.Range("C35") = Data3.TextBox1
    Plan3.Range("B36") = Ap_ef
    Plan3.Range("B37") = Pi_ef
    Data3.Repaint
    Unload Me
    DesenhaViga 50, 480, 750, 1, 150
    DesenhaFuso 50, 480, 750, 1
    Atualiza_LB_Cabos
End Sub

```

Data 4

```

Private TESTE As Integer 'Testa se o formulário foi
fechado pelo X

```

```

Private Sub btOK_4_Click()
    FUSO = 0
    EQ = 0
    If Plan2.Range("M17") < 4 Then
Plan2.Range("M17") = 4
        If Data1.ROT.Value = True Then
            Me.Hide
            Data5.Show
            Unload Me
        Else
            TESTE = 1
            Unload Me
        End If
    End Sub

```

```

Private Sub ComboBox1_Change()
    dado = ComboBox1.Text
    Nsec = Plan3.Range("C13")
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
Then
            Me.ComboBox1.Text = Mid(dado, 1,
Len(dado) - 1)
        Exit Sub
    End If
    Else
        Label2 = "x: "
        Label3 = "e_lim_sup: "
        Label4 = "e_lim_inf: "
        Exit Sub
    End If
    dado = Int(dado)
    If Int(dado) > Int(Nsec) Then
        Label2 = "x: Seção inválida"
        Label3 = "e_lim_sup: Seção inválida"
        Label4 = "e_lim_inf: Seção inválida"
        Exit Sub
    End If
    If dado <= Nsec Then
        Label2 = "x: " & Format(ep_lim_inf(dado, 0),
"0.0000")
        Label3 = "e_lim_sup: " &
Format(ep_lim_sup(dado, 1), "0.0000")
        Label4 = "e_lim_inf: " &
Format(ep_lim_inf(dado, 1), "0.0000")
        Exit Sub
    End If
End Sub

```

```

Private Sub CommandButton1_Click()
    Cabo = ListBox1.Text
    If Cabo = "" Then
        MsgBox "Selecione o cabo a ser modificado.",
vbOKOnly, "Selecione o cabo"
    End Sub

```

```

Else
    Mod_Cabo.Show
End If
End Sub

Private Sub TextBox1_Change()
    dado = TextBox1.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
            End If
            If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
                Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
            End If
            If ContaCarac(dado, ",") = 1 Then
                d = Split(dado, ",")
                If Len(d(1)) = 4 Then Me.TextBox1.Text =
Mid(dado, 1, Len(dado) - 1)
            End If
        End If
        If TextBox1 > 0 And TextBox1 <> "" Then
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            If FUSO = 1 Then DesenhaFuso_2 50, 480,
750, TextBox1
            If EQ = 1 Then DesenhaEquivalente 50, 480,
750, TextBox1
            TextBox1 = esc
        End If
    End Sub

Private Sub ToggleButton1_Change()
    FUSO = 0
    If ToggleButton1 = True Then
        If EQ = 1 Then
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaFuso_2 50, 480, 750, TextBox1
            DesenhaEquivalente 50, 480, 750,
TextBox1
        Else
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaFuso_2 50, 480, 750, TextBox1
        End If
        FUSO = 1
    Else
        FUSO = 0
        If EQ = 1 Then
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaEquivalente 50, 480, 750,
TextBox1
        Else
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaFuso_2 50, 480, 750, TextBox1
        End If
    End Sub

Private Sub ToggleButton2_Change()
    EQ = 0
    If ToggleButton2 = True Then
        If FUSO = 1 Then
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaFuso_2 50, 480, 750, TextBox1
            DesenhaEquivalente 50, 480, 750,
TextBox1
        Else
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaEquivalente 50, 480, 750,
TextBox1
        End If
        EQ = 1
    Else
        EQ = 0
        If FUSO = 1 Then
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
            DesenhaFuso_2 50, 480, 750, TextBox1
        Else
            Me.Repaint
            DesenhaViga 50, 480, 750, TextBox1, 300
            DesenhaCabos 50, 480, 750, TextBox1
        End If
    End Sub

Private Sub UserForm_Activate()
    If Plan2.Range("M17") < 3 Then
        MsgBox "Abrir primeiro janelas anteriores!"
        Unload Data4
        Data4.Hide
        Exit Sub
    End If
    Me.Repaint
    DesenhaViga 50, 480, 750, 1, 300
    DesenhaCabos 50, 480, 750, 1
    Cabo_Equivalente
    TextBox1 = esc
End Sub

Private Sub UserForm_Initialize()
    ult = Ultima_B(65, 65)
    Nsec = Plan3.Range("C13")
DesenhaEquivalente 50, 480, 750,
TextBox1
Else
Me.Repaint
DesenhaViga 50, 480, 750, TextBox1, 300
DesenhaCabos 50, 480, 750, TextBox1
End If
End Sub
Private Sub ToggleButton2_Change()
EQ = 0
If ToggleButton2 = True Then
If FUSO = 1 Then
Me.Repaint
DesenhaViga 50, 480, 750, TextBox1, 300
DesenhaCabos 50, 480, 750, TextBox1
DesenhaFuso_2 50, 480, 750, TextBox1
DesenhaEquivalente 50, 480, 750,
TextBox1
Else
Me.Repaint
DesenhaViga 50, 480, 750, TextBox1, 300
DesenhaCabos 50, 480, 750, TextBox1
DesenhaEquivalente 50, 480, 750,
TextBox1
End If
EQ = 1
Else
EQ = 0
If FUSO = 1 Then
Me.Repaint
DesenhaViga 50, 480, 750, TextBox1, 300
DesenhaCabos 50, 480, 750, TextBox1
DesenhaFuso_2 50, 480, 750, TextBox1
End If
End Sub
Private Sub UserForm_Activate()
If Plan2.Range("M17") < 3 Then
MsgBox "Abrir primeiro janelas anteriores!"
Unload Data4
Data4.Hide
Exit Sub
End If
Me.Repaint
DesenhaViga 50, 480, 750, 1, 300
DesenhaCabos 50, 480, 750, 1
Cabo_Equivalente
TextBox1 = esc
End Sub
Private Sub UserForm_Initialize()
ult = Ultima_B(65, 65)
Nsec = Plan3.Range("C13")

```

```

Data4.ListBox1.ColumnCount = 5
If ult > 1 Then Data4.ListBox1.RowSource =
"BIBLIOTECA!BM2:BQ" & ult
ult2 = Ultima_B(64, 64)
Plan2.Range("BL1", "BL" & ult2).Clear
For i = 0 To Nsec
    Plan2.Range("BL" & (i + 1)) = i
Next i
ult3 = Ultima_B(64, 64)
ComboBox1.RowSource =
"BIBLIOTECA!BL1:BL" & ult3
If Plan3.Range("E32") = "Atenção!" Then
    Label20.ForeColor = RGB(255, 0, 0)
    Label20.Visible = True
    Label21.Visible = True
End If
End Sub

```

```

Private Sub Data4_QueryClose(Cancel As Integer,
CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then
        Cancel = False
        Application.Quit
    Application.DisplayAlerts = False
    End If
End Sub

```

Modificar cabo

```

Private Sub CommandButton1_Click()
    If TextBox1 = "" Or TextBox2 = "" Or TextBox3
= "" Then
        MsgBox "Preencha todos os campos.",
vbOKOnly, "Dados incompletos"
        Exit Sub
    End If
    L = Plan3.Range("B12")
    If TextBox2.Text > (L / 2) Then
        MsgBox "xm deve ser menor que metade do
vão", vbOKOnly, "Dados incorretos"
        Exit Sub
    End If
    h = Plan3.Range("B6")
    If Abs(TextBox1.Text) >= h Or
Abs(TextBox3.Text) >= h Then
        MsgBox "Valor de excentricidade muito
elevado", vbOKOnly, "Dados incorretos"
        Exit Sub
    End If
    ult = Ultima_B(65, 65)
    Cabo = Data4.ListBox1.Text
    TextBox1 = Format(Mod_Cabo.TextBox1,
"0.0000")
    TextBox3 = Format(Mod_Cabo.TextBox3,
"0.0000")
    For i = 2 To ult
        If Plan2.Range("BM" & i) = CDb1(Cabo)
Then
            Plan2.Range("BN" & i) = TextBox1

```

```

Plan2.Range("BO" & i) =
Format(Mod_Cabo.TextBox2, "0.00")
Plan2.Range("BP" & i) = TextBox3
    End If
Next i
Unload Me
Data4.Repaint
DesenhaViga 50, 480, 750, esc, 300
DesenhaCabos 50, 480, 750, TextBox1
Cabo_Equivalente
If FUSO = 1 Then DesenhaFuso_2 50, 480, 750,
TextBox1
    If EQ = 1 Then DesenhaEquivalente 50, 480,
750, TextBox1
        Data4.TextBox1 = esc
    End Sub

```

```

Private Sub TextBox1_Change()
    dado = TextBox1.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," And
Mid(dado, Len(dado), 1) <> "-" Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, "-") > 1 And Mid(dado,
Len(dado), 1) = "-" Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 5 Then Me.TextBox1.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

```

Private Sub TextBox2_Change()
    dado = TextBox2.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox2.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")

```

```

        If Len(d(1)) = 3 Then Me.TextBox2.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

Private Sub TextBox3_Change()
    dado = TextBox3.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
And Mid(dado, Len(dado), 1) <> "," And
Mid(dado, Len(dado), 1) <> "-" Then
            Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, "-") > 1 And Mid(dado,
Len(dado), 1) = "-" Then
            Me.TextBox3.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 5 Then Me.TextBox3.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub

```

Data 5

Private TESTE As Integer "Testa se o formulário foi fechado pelo X

```

Private Sub CommandButton1_Click()
    If Plan3.Range("C23") = "Pós-Tração" And (Not
(OptionalButton1 Or OptionalButton2 Or
OptionalButton3 Or OptionalButton4 Or
OptionalButton5) Or Not (OptionalButton6 Or
OptionalButton7)) Then
        MsgBox "Dados de perda por atrito
incompletos!", vbOKOnly, "Dados insuficientes!"
        Exit Sub
    Else
        If OptionalButton1 Then u = 0.5
        If OptionalButton2 Then u = 0.3
        If OptionalButton3 Then u = 0.2
        If OptionalButton4 Then u = 0.1
        If OptionalButton5 Then u = 0.05
        If OptionalButton6 Then b = 0.01
        If OptionalButton7 Then b = TextBox1.Text
        Plan3.Range("D41") = u
        Plan3.Range("D42") = b
    End If
    If TextBox2 = "" Then

```

```

        MsgBox "Dados de perda por deformação da
ancoragem incompletos!", vbOKOnly, "Dados
insuficientes!"
        Exit Sub
    Else
        Plan3.Range("B44") = CDbI(TextBox2)
    End If
    If TextBox7 = "" Or TextBox8 = "" Then
        MsgBox "Dados de perdas progressivas
incompletos!", vbOKOnly, "Dados insuficientes!"
        Exit Sub
    Else
        Plan3.Range("B48") = CDbI(TextBox7)
        Plan3.Range("B49") = CDbI(TextBox8)

        Plan2.Range("BS2") = TextBox3
        Plan2.Range("BS3") = TextBox4
        Plan2.Range("BS4") = TextBox5
        Plan2.Range("BS5") = TextBox6
        Plan2.Range("BS6") = TextBox9
        If OptionalButton8 Then Plan2.Range("BS7") = 1
        If OptionalButton9 Then Plan2.Range("BS7") = 2
        If OptionalButton10 Then Plan2.Range("BS7") = 3
    End If
    If Not (OptionalButton11 Or OptionalButton12)
Then
        MsgBox "Dados de ancoragem incompletos!",
vbOKOnly, "Dados insuficientes!"
        Exit Sub
    Else
        If OptionalButton11 Then
            Plan3.Range("A51") = "Ativa-Ativa"
        Else
            Plan3.Range("A51") = "Ativa-Passiva"
        End If
    End If
    If Plan2.Range("M17") < 5 Then
        Plan2.Range("M17") = 5
        PERDA_ATRITO
        PERDA_ANCORAGEM
        PERDA_ENCURTAMENTO
        PERDAS_PROGRESSIVAS
        GerarPlanilhaELS ("ELS")
        ELS_Verificacao
        Plan3.Range("A39:K54").Copy
        Destination:=Plan1.Range("A39")
        TESTE = 1
        Unload Me
    End Sub

Private Sub OptionalButton10_Change()
    Fluencia_e_Retracao
End Sub

Private Sub OptionalButton7_Change()
    If OptionalButton7 Then
        TextBox1.Visible = True
        Label4.Visible = True
    Else
        TextBox1.Visible = False
        Label4.Visible = False

```

```
End If
End Sub
```

```
Private Sub OptionButton8_Change()
    Fluencia_e_Retracao
End Sub
```

```
Private Sub OptionButton9_Change()
    Fluencia_e_Retracao
End Sub
```

```
Private Sub TextBox1_Change()
    dado = TextBox1.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
            End If
            If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
                Me.TextBox1.Text = Mid(dado, 1,
Len(dado) - 1)
            End If
            If ContaCarac(dado, ",") = 1 Then
                d = Split(dado, ",")
                If Len(d(1)) = 6 Then Me.TextBox1.Text =
Mid(dado, 1, Len(dado) - 1)
            End If
        End If
    End Sub
```

```
Private Sub TextBox2_Change()
    dado = TextBox2.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox2.Text = Mid(dado, 1, Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox2.Text = Mid(dado, 1, Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.TextBox2.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub
```

```
Private Sub TextBox3_Change()
    Fluencia_e_Retracao
    dado = TextBox3.Text
    If Len(dado) = 3 Then
        Me.TextBox3.Text = Mid(dado, 1, Len(dado) - 1)
        Exit Sub
    End If
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        Then Me.TextBox3.Text = Mid(dado, 1, Len(dado) - 1)
    End If
End Sub
```

```
End If
End Sub
```

```
Private Sub TextBox4_Change()
    Fluencia_e_Retracao
    dado = TextBox4.Text
    If Len(dado) = 3 Then
        Me.TextBox4.Text = Mid(dado, 1, Len(dado) - 1)
        Exit Sub
    End If
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        Then Me.TextBox4.Text = Mid(dado, 1, Len(dado) - 1)
    End If
End Sub
```

```
Private Sub TextBox5_Change()
    Fluencia_e_Retracao
    dado = TextBox5.Text
    If Len(dado) = 3 Then
        Me.TextBox5.Text = Mid(dado, 1, Len(dado) - 1)
        Exit Sub
    End If
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        Then Me.TextBox5.Text = Mid(dado, 1, Len(dado) - 1)
    End If
End Sub
```

```
Private Sub TextBox6_Change()
    Fluencia_e_Retracao
    dado = TextBox6.Text
    If Len(dado) = 3 Then
        Me.TextBox6.Text = Mid(dado, 1, Len(dado) - 1)
        Exit Sub
    End If
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        Then Me.TextBox6.Text = Mid(dado, 1, Len(dado) - 1)
    End If
End Sub
```

```
Private Sub TextBox9_Change()
    Fluencia_e_Retracao
    dado = TextBox9.Text
    If Len(dado) > 0 Then
        If Not IsNumeric(Mid(dado, Len(dado), 1))
        And Mid(dado, Len(dado), 1) <> "," Then
            Me.TextBox9.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") > 1 And Mid(dado,
Len(dado), 1) = "," Then
            Me.TextBox9.Text = Mid(dado, 1,
Len(dado) - 1)
        End If
        If ContaCarac(dado, ",") = 1 Then
            d = Split(dado, ",")
            If Len(d(1)) = 4 Then Me.TextBox9.Text =
Mid(dado, 1, Len(dado) - 1)
        End If
    End If
End Sub
```

```

End If
End Sub

Private Sub UserForm_Activate()
If Plan2.Range("M17") < 4 Then
    MsgBox "Abrir primeiro janelas anteriores!"
    Unload Data5
    Data5.Hide
End If
End Sub

Private Sub UserForm_Initialize()
If Plan3.Range("A51") = "Ativa-Ativa" Then
OptionButton11.Value = True
If Plan3.Range("A51") = "Ativa-Passiva" Then
OptionButton12.Value = True
If Plan3.Range("C23") = "Pré-Tração" Then
    Frame1.Enabled = False
    Frame4.Enabled = False
    Frame5.Enabled = False
    OptionButton1.Enabled = False
    OptionButton2.Enabled = False
    OptionButton3.Enabled = False
    OptionButton4.Enabled = False
    OptionButton5.Enabled = False
    OptionButton6.Enabled = False
    OptionButton7.Enabled = False
End If
u = Plan3.Range("D41")
b = Plan3.Range("D42")
If u = 0.5 Then OptionButton1.Value = True
If u = 0.3 Then OptionButton2.Value = True
If u = 0.2 Then OptionButton3.Value = True
If u = 0.1 Then OptionButton4.Value = True
If u = 0.05 Then OptionButton5.Value = True
If b = 0.01 Then
    OptionButton6.Value = True
Else
    If b <> 0 And b <> "-" Then
        OptionButton7.Value = True
        TextBox1 = b
    End If
End If
TextBox2 = Plan3.Range("B44")
TextBox3 = Plan2.Range("BS2")
TextBox4 = Plan2.Range("BS3")
TextBox5 = Plan2.Range("BS4")
TextBox6 = Plan2.Range("BS5")
TextBox9 = Plan2.Range("BS6")
Tipo = Plan2.Range("BS7")
If Tipo = 1 Then OptionButton8.Value = True
If Tipo = 2 Then OptionButton9.Value = True
If Tipo = 3 Then OptionButton10.Value = True
TextBox7 = Plan3.Range("B48")
TextBox8 = Plan3.Range("B49")
End Sub

Private Sub Data5_QueryClose(Cancel As Integer,
CloseMode As Integer)
If CloseMode = vbFormControlMenu Then

```

```

Cancel = False
Application.Quit
Application.DisplayAlerts = False
End If
End Sub

```

Data 6

```

Private Sub CommandButton1_Click()
Static isWorking As Boolean
Nsec = Plan3.Range("C13")
L = Plan3.Range("B12")
Ac = Plan3.Range("B7")
Ic = Plan3.Range("D9")
'ENTRADA
If TextBox1 = "" Or TextBox2 = "" Or TextBox3
= "" Or TextBox4 = "" Or TextBox5 = "" Or
TextBox6 = "" Or TextBox7 = "" Or TextBox8 = ""
Or ((Not OptionButton1) And (Not
OptionButton2)) Then
    MsgBox "Dados insuficientes!", vbOKOnly,
"Dados insuficientes!"
    Exit Sub
End If
If OptionButton1 Then Plan2.Range("CA2") = 1
If OptionButton2 Then Plan2.Range("CA2") = 0
Plan2.Range("CA3") = TextBox1.Text
Plan2.Range("CA4") = TextBox2.Text
Plan2.Range("CA5") = TextBox3.Text
Plan2.Range("CA6") = TextBox4.Text
Plan2.Range("CA7") = TextBox5.Text
Plan2.Range("CA8") = TextBox6.Text
Plan2.Range("CA9") = TextBox7.Text
Plan2.Range("CA10") = TextBox8.Text
'CONHECIDOS
Plan2.Range("CA14") = Plan3.Range("B6") * 100
Plan2.Range("CA15") =
Plan3.Range("B12").Text
Plan2.Range("CA16") =
Cdbl(Plan3.Range("B36").Text)
Plan2.Range("CA18") =
Plan3.Range("B16").Text
Plan2.Range("CA19") =
Cdbl(Plan3.Range("E19").Text)
Plan2.Range("CA20") = 210000
Plan2.Range("CA21") =
Plan3.Range("E20").Text
Plan2.Range("CA22") = Plan3.Range("B7") * (100 ^ 2)
Plan2.Range("CA23") = Plan3.Range("D9") * (100 ^ 4)
Plan2.Range("CA24") = (Plan3.Range("B6") -
Plan3.Range("D8")) * 100
Plan2.Range("CA25") = Plan3.Range("D8") * 100
Plan2.Range("CA29") = Plan2.Range("BK4") / 10
'VARIA COM A SEÇÃO (Msd, Pinf, eta)
Cabo_Equivalente
PERDA_ATRITO
PERDA_ANCORAGEM
PERDA_ENCURTAMENTO
PERDAS_PROGRESSIVAS

```

```

GerarPlanilhaELU ("ELU")
ActiveSheet.Range("A1:G4").Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlCenter
    .MergeCells = True
End With
With Selection.Font
    .Size = 24
    .Bold = True
End With
ActiveSheet.Range("A1") = "ELU -
VERIFICAÇÃO"
ActiveSheet.Range("E7:G8").Select
With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlCenter
    .MergeCells = True
    .WrapText = True
End With
With Selection.Font
    .Size = 12
    .Bold = True
    .Color = vbRed
End With
ActiveSheet.Range("E7") = "OBS: Decalagem
não foi considerada."

k = 0
For i = 0 To Nsec
    eta = 1 + (EQUIVALENTE(i, 0) ^ 2) * Ac / Ic
    Plan2.Range("CA28") = eta
    Plan2.Range("CA17") = PROGRESSIVAS(i)
    If i = 0 Then
        x = L / 2
    Else
        x = ((i - 1) * L) / (Nsec - 1)
    End If
    VetorMomentos (x)
    MSd = Comb_ELU
    Plan2.Range("CA13") = MSd
    If Not isWorking Then
        isWorking = True
        Plan2.Range("CD2") = 0.1
        Plan2.Range("CD21").GoalSeek
Goal:=MSd, ChangingCell:=Plan2.Range("CD2")
        isWorking = False
    End If
    'Plan2.Range("CD2") = 0.1
    'Plan2.Range("CD21").GoalSeek Goal:=MSd,
ChangingCell:=Plan2.Range("CD2")
    MsgBox "Seção: " & i & " Msd: " & MSd & "
Eta: " & eta & " Pinf: " & PROGRESSIVAS(i) & "
Mrd: " & Plan2.Range("CD21") & " x/d: " &
Plan2.Range("CD2")
    Sheets("ELU").Select
    ActiveSheet.Range("A" & (i + k + 5), "G" & (i
+ k + 5)).Merge
    ActiveSheet.Range("A" & (i + k +
5)).Interior.Color = RGB(220, 230, 241)

    If i = 0 Then
        ActiveSheet.Range("A" & (i + k + 5)) =
"MEIO DO VÃO"
        x = L / 2
    Else
        ActiveSheet.Range("A" & (i + k + 5)) =
"SEÇÃO " & i
        x = ((i - 1) * L) / (Nsec - 1)
    End If
    ActiveSheet.Range("A" & (i + k + 6)) =
"Pinf:"
    ActiveSheet.Range("B" & (i + k + 6)) =
Application.WorksheetFunction.RoundUp(PROGR
ESSIVAS(i), 3)
    ActiveSheet.Range("C" & (i + k + 6)) = "kN"
    ActiveSheet.Range("A" & (i + k + 7)) =
"Msd=Mrd:"
    ActiveSheet.Range("B" & (i + k + 7)) =
Application.WorksheetFunction.RoundUp(MSd, 3)
    ActiveSheet.Range("C" & (i + k + 7)) =
"kN.m"
    ActiveSheet.Range("A" & (i + k + 8)) = "x/d:"
    ActiveSheet.Range("B" & (i + k + 8)) =
Application.WorksheetFunction.RoundUp(Plan2.R
ange("CD2"), 3)
    ActiveSheet.Range("A" & (i + k + 9)) = "As:"
    ActiveSheet.Range("B" & (i + k + 9)) =
Application.WorksheetFunction.RoundUp(Plan2.R
ange("CD19"), 3)
    ActiveSheet.Range("C" & (i + k + 9)) = "cm2"
    ActiveSheet.Range("A" & (i + k + 10)) =
"As,min:"
    ActiveSheet.Range("B" & (i + k + 10)) =
Plan2.Range("CD22")
    ActiveSheet.Range("C" & (i + k + 10)) =
"cm2"
    k = k + 6
Next i
Sheets("ELU").Select
ActiveSheet.Columns.AutoFit
ActiveSheet.Range("B6", "B" & (((Nsec + 1) *
10) + 4)).HorizontalAlignment = xlCenter
Plan3.Range("A56:K57").Copy
Destination:=Plan1.Range("A56")
Unload Me
End Sub

Private Sub CommandButton2_Click()

    MSd = Plan2.Range("CA13")
    Plan2.Range("CD21").GoalSeek Goal:=MSd,
ChangingCell:=Plan2.Range("CD2")

End Sub

Private Sub UserForm_Activate()
If Plan2.Range("M17") < 5 Then
    MsgBox "Abrir primeiro janelas anteriores!"
    Unload Data6
    Data6.Hide

```

```

End If
End Sub

Private Sub UserForm_Initialize()

    TextBox1 = Plan2.Range("CA3")
    TextBox2 = Plan2.Range("CA4")
    TextBox3 = Plan2.Range("CA5")
    TextBox4 = Plan2.Range("CA6")
    TextBox5 = Plan2.Range("CA7")
    TextBox6 = Plan2.Range("CA8")
    TextBox7 = Plan2.Range("CA9")
    TextBox8 = Plan2.Range("CA10")

    If Plan2.Range("CA2") = 1 Then OptionButton1
= True
    If Plan2.Range("CA2") = 0 Then OptionButton2
= True

End Sub

```

Data 7

```

Private Sub UserForm_Initialize()
Nsec = Plan3.Range("C13")
L = Plan3.Range("B12")
MCFcritico = 0
xcritico = 0
SecCrit = 1

For secas = 0 To Nsec

    If secas = 0 Then
        x = L / 2
    Else
        x = (L / (Nsec - 1)) * (secas - 1)
    End If
    Call Functions.MomentoCF(x)
    Mcf = MomentoCFTotal
    If Mcf > MCFcritico Then
        MCFcritico = Mcf
        xcritico = x
        SecCrit = secas
    End If

Next secas

Plan2.Range("DE1") = SecCrit
Plan2.Range("DE2") = xcritico 'm
Plan2.Range("DE3") = MCFcritico 'kN.m
Label4 = "x = " & xcritico & " m"
Label15 = " "

    Set Planilha5 = Sheets("ELU")
    Asnec = Planilha5.Range("B" & (9 + 7 *
SecCrit))
    Asmin = Planilha5.Range("B" & (10 + 7 *
SecCrit))

```

```

If Asmin > Asnec Then
AA = Asmin
Else
AA = Asnec
End If
Label6 = AA

```

```

    ComboBox2.RowSource =
"BIBLIOTECA!CP3:CP10"
    ComboBox3.RowSource =
"BIBLIOTECA!CP3:CP7"
    ComboBox4.RowSource =
"BIBLIOTECA!CW3:CW5"
Sheets("PARCIAL").Select
Plan6.Range("B10") = 0
Plan6.Range("B23") = 0
End Sub

```

```

Private Sub ComboBox2_Change()
dado = ComboBox2.Text / 10
NBarras2 = Label6 / ((3.14159265359 * dado ^ 2 /
4))
NBarras = ArredondarParaCima(CDbl(NBarras2))
If NBarras < 2 Then
    NBarras = 2
End If
Plan6.Range("N30") = NBarras
Label13 = NBarras & " " & ChrW(632) & " " &
ComboBox2.Text
Label28 = Format(NBarras * (3.14159265359 *
dado ^ 2 / 4), "0.00")
Plan2.Range("DE4") = Format(NBarras *
(3.14159265359 * dado ^ 2 / 4), "0.00") 'cm2
End Sub

```

```

Private Sub CommandButton2_Click()

Nramos = ComboBox4
dado = ComboBox2.Text / 10
bw = Plan2.Range("CA3")
cobrimento = TextBox4
BitolaEstribo = ComboBox3 / 10
BitolaLong = ComboBox2 / 10
Dagregado = TextBox3 / 10
LarguraLivre = bw - 2 * cobrimento - Nramos *
BitolaEstribo

```

```

EspacoMinimoH = 2
If EspacoMinimoH < BitolaLong Then
    EspacoMinimoH = BitolaLong
End If
If EspacoMinimoH < 1.2 * Dagregado Then
    EspacoMinimoH = 1.2 * Dagregado
End If

```

```

NBarras2 = Label6 / ((3.14159265359 * dado ^ 2 /
4))
TotBarras =
ArredondarParaCima(CDbl(NBarras2))

```



```

If TotBarras < 2 Then
  TotBarras = 2
End If
NBCamada = TotBarras
For i = 1 To TotBarras
  If (NBCamada * BitolaLong + (NBCamada - 1) *
EspacoMinimoH) <= LarguraLivre Then
    NBCamada = NBCamada
  Else
    NBCamada = NBCamada - 1
  End If
Next i
NCamadasCheias = Int(TotBarras / NBCamada)
NBarrasUlt = TotBarras - NCamadasCheias *
NBCamada
If NBarrasUlt = 0 Then
  NCamadas = NCamadasCheias
  Label15 = NCamadasCheias & " camadas de " &
NBCamada & ChrW(632) & ComboBox2
Else
  NCamadas = NCamadasCheias + 1
  Label15 = NCamadasCheias & " camadas de " &
NBCamada & ChrW(632) & ComboBox2 & " + 1
camada de " & NBarrasUlt & ChrW(632) &
ComboBox2
End If

EspacoMinimoV = 2
If EspacoMinimoV < BitolaLong Then
  EspacoMinimoV = BitolaLong
End If
If EspacoMinimoV < 0.5 * Dagregado Then
  EspacoMinimoV = 0.5 * Dagregado
End If
'
'
' CÁLCULO DA ALTURA ÚTIL DA
ARMADURA PASSIVA
'
'
h = 100 * Plan3.Range("B6")
ReDim d(1 To NCamadas)
ReDim nd(1 To NCamadas)
auxiliar = 0
For j = 1 To NCamadas
  d(j) = (h - cobrimento - BitolaEstribo - BitolaLong
/ 2) - (j - 1) * (EspacoMinimoV + BitolaLong)
  If NBarrasUlt = 0 Then
    nd(j) = NBCamada * d(j)
  Else
    nd(j) = NBCamada * d(j)
    nd(NCamadas) = NBarrasUlt * d(NCamadas)
  End If
  auxiliar = auxiliar + nd(j)
Next j
dfinal = auxiliar / TotBarras
Plan6.Range("B14") = dfinal
Plan2.Range("DE5") = dfinal 'cm
If NCamadas > 1 Then
  Plan6.Range("N26") = NCamadas - 1

  alturaPenultCamada = h - d(NCamadas - 1)
  Plan6.Range("O26") = alturaPenultCamada
  Plan6.Range("N27") = NCamadas
  alturaUltCamada = h - d(NCamadas)
  Plan6.Range("O27") = alturaUltCamada
End If
'
'
' CÁLCULO DA ÁREA CRÍTICA
'
'
'Distância máxima de concreto de envolvimento
dmaxima = 7.5 * BitolaLong
'Distâncias horizontais
'Distância horizontal do eixo da barra extrema até a
face lateral
d1 = cobrimento + BitolaEstribo + BitolaLong / 2
  If d1 > dmaxima Then
    d1 = dmaxima
  End If
'Distância horizontal entre eixos das barras dividida
por dois(para o caso de apenas uma camada ou
camadas cheias)
d2 = ((bw - 2 * cobrimento - 2 * BitolaEstribo -
BitolaLong) / (NBCamada - 1)) / 2
  If d2 > dmaxima Then
    d2 = dmaxima
  End If
'Distância horizontal entre eixos das barras da
última camada incompleta dividida por dois (caso
haja mais de uma camada)
If NBarrasUlt = 1 Then
  d3 = bw - cobrimento - BitolaEstribo -
BitolaLong / 2
End If
If NBarrasUlt > 1 Then
  d3 = ((bw - 2 * cobrimento - 2 * BitolaEstribo -
BitolaLong) / (NBarrasUlt - 1)) / 2
End If
  If d3 > dmaxima Then
    d3 = dmaxima
  End If
'Distâncias verticais
'Distância vertical do eixo das barras da primeira
camada até a face é igual a d1
'Distância vertical acima do eixo das barras da
última camada é igual a dmaxima
'
' Caso de apenas uma camada com até duas barras
If NCamadas = 1 And NBCamada < 3 Then
  Acr = NBCamada * (d1 + d2) * (d1 + dmaxima)
'canto primeira camada
End If
' Caso de apenas uma camada com mais que duas
barras
If NCamadas = 1 And NBCamada > 2 Then
  Acr = 2 * (d1 + d2) * (d1 + dmaxima) +
(NBCamada - 2) * 2 * d2 * (d1 + dmaxima) 'canto
primeira camada
End If

```

```

'Caso de duas ou mais camadas completas
If NCamadas > 1 And NBarrasUlt = 0 Then
  Acr = bw * (alturaUltCamada + dmaxima)
End If
'Caso de duas ou mais camadas com a última
camada contendo até duas barras
If NCamadas > 1 And NBarrasUlt < 3 Then
  Acr = bw * (alturaPenultCamada + dmaxima) +
  NBarrasUlt * (d1 + d2) * ((alturaUltCamada +
  dmaxima) - (alturaPenultCamada + dmaxima))
End If
'Caso de duas ou mais camadas com a última
camada contendo mais que duas barras
If NCamadas > 1 And NBarrasUlt > 2 Then
  Acr = bw * (alturaPenultCamada + dmaxima) +
  (2 * (d1 + d2) + (NBCamada - 2) * 2 * d2) *
  ((alturaUltCamada + dmaxima) -
  (alturaPenultCamada + dmaxima))
End If
Label35 = Acr
Plan6.Range("B35") = Acr

If Abs(TotBarras) < Abs(Nramos) Then
  MsgBox "Número de barras longitudinais inferior
  ao número de ramos. Reduza o número de ramos ou
  o diâmetro da armadura longitudinal.",
  vbInformation, "Informações"
End If
End Sub

Private Sub CommandButton3_Click()
'Calcula tensão de tração
L = Plan3.Range("B12")
fator = Plan2.Range("O17")
aderencia = Plan2.Range("CA2")
h = 100 * Plan3.Range("B6")
SecCrit = Plan2.Range("DE1")

'Momento na combinação frequente
xcritico = Plan2.Range("DE2")
Call Functions.MomentoCF(xcritico)

'Dados para Cálculo do Epré-alongamento
Ep = Plan3.Range("E20") / 10 'kN/cm²
Plan6.Range("B18") = Ep
Ecs = Plan3.Range("E19") / 10 'kN/cm²
Esi = Plan2.Range("CA20") / 10 'kN/cm²
alphap = Ep / Ecs
EpSobreEs = Ep / Esi
Plan6.Range("E11") = EpSobreEs
Ap = Plan3.Range("B36") 'cm²
Plan6.Range("B16") = Ap 'cm²
Plan2.Range("DE7") = Ap 'cm²
If fator = "1,5" Then
  bc = Plan2.Range("CA3")
Else
  bc = Plan2.Range("CA4")
End If
Plan6.Range("B12") = bc
cg = 100 * Plan3.Range("D8") 'cm
yct = h - cg 'cm
Plan6.Range("E12") = yct
Set Planilha4 = Sheets("ELS")
dp = h - cg + 100 * Planilha4.Range("B" & (24 +
14 * SecCrit)) 'cm
Plan6.Range("B19") = dp
Plan2.Range("DE8") = dp
rop = Ap / (bc * dp)
excp = 100 * Planilha4.Range("B" & (24 + 14 *
SecCrit)) 'cm
Ac = 10000 * Plan3.Range("B7") 'cm²
Ic = 100000000 * Plan3.Range("D9") 'cm4
eta = 1 + (excp ^ 2) * Ac / Ic
PinfSecao = Planilha4.Range("B" & (30 + 14 *
SecCrit))
'Cálculo do Epré-alongamento
If Plan3.Range("C23") = "Pós-Tração" Then
  Eprealongamento = ((PinfSecao / (Ep * Ap)) * (1
+ alphap * rop * eta))
Else
  Eprealongamento = (PinfSecao / (Ep * Ap))
End If
Plan6.Range("B17") = Eprealongamento

'Cálculo da força de protensão devida ao pré-
alongamento
Ninfpre = Ap * Eprealongamento * Ep
Plan6.Range("B11") = Ninfpre

'Cálculo do deltaM
deltaM = (Plan6.Range("B8") - Ninfpre * excp)
'kN.cm
Plan6.Range("B9") = deltaM

'Cálculo de sigmas
'xparcial é uma incógnita a ser encontrada, inicia
com valor de H
xparcial = 0.2 * h
Plan6.Range("B10") = xparcial
alphaE = Esi / Ecs
Plan6.Range("B13") = alphaE
d = Plan6.Range("B14")
Asparcial = Label28

Plan6.Range("B15") = Asparcial
dsLinha = Plan2.Range("CA7")
Plan6.Range("E14") = dsLinha
AsLinha = Plan2.Range("CA8")
Plan6.Range("E15") = AsLinha

'Atingir meta (Ret) ou (T com LN na mesa)
If aderencia = 0 Then
  Plan6.Range("B22").Select

```

```
Plan6.Range("B22").GoalSeek Goal:=0,
ChangingCell:=Plan6.Range("B10")
End If
```

```
If aderencia = 1 Then
Plan6.Range("E22").Select
Plan6.Range("E22").GoalSeek Goal:=0,
ChangingCell:=Plan6.Range("B10")
End If
```

```
hf = Plan2.Range("CA5")
Plan6.Range("B26") = hf
xparcialT = 0.2 * h
Plan6.Range("B23") = xparcialT
bw = Plan2.Range("CA3")
Plan6.Range("B24") = bw
bf = Plan2.Range("CA4")
Plan6.Range("B25") = bf
```

```
'Atingir meta T com LN na alma
```

```
If aderencia = 0 Then
Plan6.Range("B30").Select
Plan6.Range("B30").GoalSeek Goal:=0,
ChangingCell:=Plan6.Range("B23")
End If
```

```
If aderencia = 1 Then
Plan6.Range("E30").Select
Plan6.Range("E30").GoalSeek Goal:=0,
ChangingCell:=Plan6.Range("B23")
End If
```

```
'Testa qual Xparcial e sigmaS usar'
```

```
xparcialteste = Plan6.Range("B10")
xparcialverdadeiro = 0
sigmaSverdadeira = 0
If aderencia = 0 Then
If fator <> 1.5 And xparcialteste > hf Then
xparcialverdadeiro = Plan6.Range("B23")
sigmaSverdadeira = Plan6.Range("B28")
Plan6.Range("B33") = xparcialverdadeiro
Plan6.Range("C33") = "LN na alma"
Plan6.Range("B34") = sigmaSverdadeira
Label20 = Format(sigmaSverdadeira, "0.00")
Else
xparcialverdadeiro = Plan6.Range("B10")
sigmaSverdadeira = Plan6.Range("B20")
Plan6.Range("B33") = xparcialverdadeiro
Plan6.Range("B34") = sigmaSverdadeira
Label20 = Format(sigmaSverdadeira, "0.00")
End If
If fator <> 1.5 And xparcialteste <= hf Then
Plan6.Range("C33") = "LN na mesa"
End If
```

```
End If
aux = xparcialteste - hf
If aderencia = 1 Then
If fator <> 1.5 And aux > 0 Then
```

```
xparcialverdadeiro = Plan6.Range("B23")
sigmaSverdadeira = Plan6.Range("E28")
Plan6.Range("B33") = xparcialverdadeiro
Plan6.Range("C33") = "LN na alma"
Plan6.Range("B34") = sigmaSverdadeira
Label20 = Format(sigmaSverdadeira, "0.00")
Else
```

```
xparcialverdadeiro = Plan6.Range("B10")
sigmaSverdadeira = Plan6.Range("E20")
Plan6.Range("B33") = xparcialverdadeiro
Plan6.Range("C33") = " "
Plan6.Range("B34") = sigmaSverdadeira
Label20 = Format(sigmaSverdadeira, "0.00")
End If
```

```
If fator <> 1.5 And aux <= 0 Then
Plan6.Range("C33") = "LN na mesa"
End If
End If
```

```
'Calcula abertura de fissuras
```

```
Acr = Plan6.Range("B35") 'cm²
Bitola = ComboBox2 / 10 'cm
NBarras = Plan6.Range("N30")
roR = (NBarras * (3.141592654 * Bitola ^ 2) / 4) /
Acr 'adimensional
eta1 = 2.25 'Considerarei apenas barra nervurada
'adimensional
```

```
fckk = Plan3.Range("B16") 'MPa
fctmm = (0.3 * fckk ^ (2 / 3)) / 10 'kN/cm²
wk1 = 10 * (Bitola / (12.5 * eta1)) *
(sigmaSverdadeira / Esi) * (3 * sigmaSverdadeira /
fctmm)
```

```
wk2 = 10 * (Bitola / (12.5 * eta1)) *
(sigmaSverdadeira / Esi) * (4 / roR + 45)
wk = wk1
```

```
If wk2 < wk Then
wk = wk2
```

```
End If
Plan6.Range("B36") = Format(wk, "0.000")
Label36 = Format(wk, "0.000")
```

```
'Verificação
```

```
If wk > 0.2 Then
Plan6.Range("B37") = "Não passa!"
Label37 = "Não passa!"
```

```
Else
Plan6.Range("B37") = "Ok!"
Label37 = "Ok!"
```

```
End If
End Sub
```

```
Private Sub CommandButton1_Click()
```

```
Unload Me
End Sub
```

Data 8

```
Private Sub UserForm_Initialize()
Set Planilha6 = Sheets("PARCIAL")
```

```

Set Planilha4 = Sheets("ELS")
.....
Plan7.Range("A11") = "Verificação da biela
comprimida e armadura transversal mínima:"
Plan2.Range("CP19") = "NAO"
L = Plan3.Range("B12") 'm
'Armazena dados dos apoios
Plan2.Range("CT18") = 0 'm
Plan2.Range("CT19") = 0 'm
Plan2.Range("CT21") = 0 'm
Plan2.Range("CT22") = 0 'm
ComboBox11.RowSource =
"BIBLIOTECA!DA3:DA6"
ComboBox11.Visible = False
Label148.Visible = False
Label149.Visible = False
Label151.Visible = False
Label152.Visible = False
Label153.Visible = False
Label117.Visible = False
Label27.Visible = False
Label25.Visible = False
'Preenche início da 1ªfaixa e final da 3ªfaixa
xi1 = 0 'cm
Label130 = xi1 'cm
Plan7.Range("B23") = xi1 'cm
xf3 = 100 * L 'cm
Label133 = xf3 'cm
Plan7.Range("B44") = xf3 'cm
.....
Data8.Height = 399
Frame1.Visible = False
Frame7.Visible = False
Image1.Visible = False
Plan7.Range("B18") = " "
ComboBox1.RowSource =
"BIBLIOTECA!CW3:CW5"
ComboBox3.RowSource =
"BIBLIOTECA!CW3:CW5"
ComboBox5.RowSource =
"BIBLIOTECA!CW3:CW5"
ComboBox8.RowSource =
"BIBLIOTECA!CY3:CY4"
ComboBox9.RowSource =
"BIBLIOTECA!CP15:CP16"
ComboBox10.RowSource =
"BIBLIOTECA!CP15:CP16"
'Cálculo do da Carga equivalente ao Pinfinito
TotalFios = Plan3.Range("C35")
ultC = Ultima_B(62, 62)
If TotalFios = 0 Or ultC = 10 Then Exit Sub
num_cabos = Plan2.Range("BJ" & ultC)
somotorioei = 0
somotorioem = 0
For i = 1 To num_cabos
    num_fios = Plan2.Range("BK" & (i + 10))
    linha = i + 1
    ei = Plan2.Range("BN" & linha)
    em = Plan2.Range("BP" & linha)
    somatorioei = somatorioei + ei * num_fios

```

```

somotorioem = somatorioem + em *
num_fios
Next i
ei_eq = somatorioei / TotalFios
em_eq = somatorioem / TotalFios
Plan2.Range("CT4") = em_eq
exc_eq = em_eq - ei_eq
Pinfinito = Planilha4.Range("B30") ' Pinfinito no
meio do vão (kN)
qALIV = 8 * Pinfinito * exc_eq / L ^ 2
Label2 = Format(qALIV, "0.00")
Plan7.Range("B6") = Label2
ult3 = Ultima_B(64, 64)
ComboBox7.RowSource =
"BIBLIOTECA!BL2:BL" & ult3
.....
'Compara os cortantes nos apoios e escolhe o maior
absoluto
Call Functions.Cortante_Biapoiaa_Apoio1(0)
Call Functions.Cortante_Biapoiaa_Apoio2(L)
Vsd1 = Abs(Plan2.Range("CT2"))
Vsd2 = Abs(Plan2.Range("CT3"))
Vsdmax = Vsd1
If Vsd2 > Vsdmax Then
    Vsdmax = Vsd2
End If
Plan7.Range("B16") = Format(Vsdmax, "0.00")
Label99 = Format(Vsdmax, "0.00")
.....
End Sub

Private Sub OptionButton1_Click()
ComboBox11.Visible = False
Label148.Visible = False
Label149.Visible = False
Label151.Visible = True
Label152.Visible = True
Label153.Visible = True
Label117.Visible = True
Label27.Visible = True
Label25.Visible = True
Plan2.Range("CY7") = "MOD1"
Plan7.Range("A11") = "Modelo I - Verificação da
biela comprimida (45°) e armadura transversal
mínima:"
End Sub

Private Sub OptionButton2_Click()
ComboBox11.Visible = True
Label148.Visible = True
Label149.Visible = True
Label151.Visible = False
Label152.Visible = False
Label153.Visible = False
Label117.Visible = False
Label27.Visible = False
Label25.Visible = False
Plan2.Range("CY7") = "MOD2"
Plan7.Range("A11") = "Modelo II - Verificação da
biela comprimida e armadura transversal mínima"

```

```

End Sub

Private Sub ComboBox11_Change()
tetagraus = ComboBox11
teta = 3.14159265359 * tetagraus / 180
Plan2.Range("CY10") = teta
Plan2.Range("Cz10") = tetagraus
Plan7.Range("A11") = "Modelo II - Verificação da
biela comprimida ( " & tetagraus & "°) e armadura
transversal mínima"
End Sub

Private Sub TextBox1_Change()
Set Planilha6 = Sheets("PARCIAL")
Set Planilha4 = Sheets("ELS")
.....
L = Plan3.Range("B12") 'm
Nsec = Plan3.Range("C13")
MSdmax = 0
For secas = 0 To Nsec
If secas = 0 Then
x = L / 2
Else
x = (L / (Nsec - 1)) * (secas - 1)
End If
Call Functions.VetorMomentos(x)
Momentoauxiliar = Plan2.Range("CT5")
If Abs(Momentoauxiliar) > Abs(MSdmax) Then
MSdmax = Momentoauxiliar
secaomax = secas
xmax = x
End If
Next secas
Plan2.Range("CT5") = MSdmax 'kN.m
.....
modelo = Plan2.Range("CY7")
teta = Plan2.Range("CY10") 'rad
ultC = Ultima_B(62, 62)
bw = Plan2.Range("CA3") 'cm
Plan7.Range("B13") = bw
num_cabos = Plan2.Range("BJ" & ultC)
bainha = TextBox1 'mm
Plan7.Range("B12") = bainha
If bainha / 10 > bw / 8 Then
bwEfet = bw - num_cabos * bainha / 10 'cm
Else
bwEfet = bw
End If
Plan7.Range("B14") = Format(bwEfet, "0.00")
Label30 = Format(bwEfet, "0.00")
fck = Plan3.Range("B16") 'MPa
altura = Plan3.Range("B6") * 100 'cm
Plan7.Range("B15") = altura
d = Planilha6.Range("B14") 'cm
If modelo = "MOD1" Then
VRd2 = 0.27 * (1 - fck / 250) * (0.1 * fck / 1.4) *
bwEfet * d
Else
VRd2 = 0.54 * (1 - fck / 250) * (0.1 * fck / 1.4) *
bwEfet * d * (1 / Tan(teta)) * (Sin(teta)) ^ 2

```

```

End If
Plan7.Range("B17") = VRd2
Label112 = VRd2
Vsdmax = Plan7.Range("B16")
If Abs(VRd2) >= Abs(Vsdmax) Then
Label105 = "Biela comprimida OK"
Plan7.Range("B18") = "Biela comprimida OK"
Else
Label105 = "Ruina da biela comprimida!!!"
Plan7.Range("B18") = "Ruina da biela
comprimida!!!"
End If
.....
'Calcula M0
Pinfinito = Planilha4.Range("B" & (30 + 14 *
secaomax)) 'Pinfinito no na seção de MSdmax (kN)
exc = Planilha4.Range("B" & (24 + 14 *
secaomax)) 'm
Wi = Plan3.Range("E10") 'm³
a = Plan3.Range("B7") ' m²
M0 = 0.9 * Pinfinito * (exc + Wi / a) 'kN.m
Plan2.Range("CT7") = M0
.....
'Calcula Vc MODELO 1
fck = Plan3.Range("B16") 'MPa
auxiliar = (1 + M0 / MSdmax)
If auxiliar > 2 Then
auxiliar = 2
End If
Vc0 = 0.09 * (1000) * (fck ^ (2 / 3)) * (bwEfet /
100) * (d / 100) 'kN
Vc = Vc0 * auxiliar 'kN
Plan2.Range("CT8") = Vc0
Plan2.Range("CT6") = Vc
If Plan2.Range("CY7") = "MOD1" Then
Label152 = Vc
End If
.....
End Sub

Private Sub ComboBox8_Change()
Set Planilha6 = Sheets("PARCIAL")
.....
'Calcula a armadura mínima transversal
fckk = Plan3.Range("B16") 'MPa
ftmm = (0.3 * fckk ^ (2 / 3)) / 10 'kN/cm²
bw = Plan2.Range("CA3") 'cm
If ComboBox8 = "CA50" Then
fywk = 50 'kN/cm²
ComboBox2.RowSource =
"BIBLIOTECA!CP4:CP7"
ComboBox4.RowSource =
"BIBLIOTECA!CP4:CP7"
ComboBox6.RowSource =
"BIBLIOTECA!CP4:CP7"
Else
fywk = 50 'kN/cm²
ComboBox2.RowSource =
"BIBLIOTECA!CP3:CP3"

```

```

    ComboBox4.RowSource =
"BIBLIOTECA!CP3:CP3"
    ComboBox6.RowSource =
"BIBLIOTECA!CP3:CP3"
End If
's foi tomado como 100cm
'alpha foi tomado como 90°, e portanto sen(alpha) =
1
Aswmin = 0.2 * (fctmm / fywk) * bw * 100 * 1
Plan7.Range("B19") = Format(Aswmin, "0.00") '
cm²/m
Plan2.Range("CT27") = Aswmin
Label24 = Format(Aswmin, "0.00") ' cm²
.....
'Calcula VSdAsmin MODELO 1
d = Planilha6.Range("B14") 'cm
Vc = Plan2.Range("CT6") 'kN
VSdAsmin = Aswmin * 0.9 * (d / 100) * fywk /
1.15 + Vc 'kN
Plan7.Range("B20") = VSdAsmin 'kN
Label27 = VSdAsmin 'kN
End Sub

Private Sub CommandButton8_Click()
Frame1.Visible = True
Image1.Visible = True
Data8.Height = 518.25
End Sub

Private Sub CommandButton9_Click()
Frame1.Visible = False
Image1.Visible = False
Data8.Height = 399
End Sub

Private Sub ComboBox7_Change()
dado = ComboBox7.Text
Nsec = Plan3.Range("C13")
L = Plan3.Range("B12")
xExibido = 0
If dado = 0 Then
    xExibido = L / 2
End If
If dado = 1 Then
    xExibido = 0
End If
If dado > 1 Then
    xExibido = (dado - 1) * L / (Nsec - 1)
End If
Label18 = "x: " & Format(xExibido, "0.00") &
" m"
Plan7.Range("B7") = Format(xExibido, "0.00")
Call Functions.VetorCortantes(xExibido)
Label8 = Plan7.Range("B8")
Label9 = Plan7.Range("B9")
Label14 = Plan7.Range("B10")
End Sub

Private Sub CommandButton10_Click()
Frame7.Visible = True

TextBox9.Visible = False
TextBox10.Visible = False
TextBox11.Visible = False
TextBox12.Visible = False
Label135.Visible = False
Label136.Visible = False
Label137.Visible = False
Label138.Visible = False
Label140.Visible = False
Label141.Visible = False
Label142.Visible = False
Label143.Visible = False
End Sub

Private Sub CommandButton11_Click()
Frame7.Visible = False
End Sub

Private Sub ComboBox9_Change()
Plan2.Range("CT17") = ComboBox9
If ComboBox9 = "Direto" Then
    Label145 = "Apoio1 - Reduz VSd"
    TextBox9.Visible = True
    TextBox10.Visible = True
    Label135.Visible = True
    Label136.Visible = True
    Label137.Visible = True
    Label138.Visible = True
Else
    Label145 = "Ap1 - Não reduz VSd"
    TextBox9.Visible = False
    TextBox10.Visible = False
    Label135.Visible = False
    Label136.Visible = False
    Label137.Visible = False
    Label138.Visible = False
End If
End Sub

Private Sub ComboBox10_Change()
Plan2.Range("CT20") = ComboBox10
If ComboBox10 = "Direto" Then
    Label146 = "Apoio2 - Reduz VSd"
    TextBox11.Visible = True
    TextBox12.Visible = True
    Label140.Visible = True
    Label141.Visible = True
    Label142.Visible = True
    Label143.Visible = True
Else
    Label146 = "Ap2 - Não reduz VSd"
    TextBox11.Visible = False
    TextBox12.Visible = False
    Label140.Visible = False
    Label141.Visible = False
    Label142.Visible = False
    Label143.Visible = False
End If
End Sub

```

```

Private Sub CommandButton7_Click()
MsgBox "As reduções indicadas no item 17.4.1.2.1
da ABNT NBR 6118:2014 não são aplicadas á
verificação da compressão diagonal do concreto.
Também não são utilizados em casos de apoios
indiretos."
End Sub

Private Sub CommandButton6_Click()
Plan2.Range("CP19") = "SIM"
L = Plan3.Range("B12") 'm
'Armazena dados dos apoios
If ComboBox9 = "Direto" And ComboBox10 =
"Direto" Then
Plan2.Range("CT18") = TextBox9 / 100 'm
Plan2.Range("CT19") = TextBox10 / 100 'm
Plan2.Range("CT21") = TextBox11 / 100 'm
Plan2.Range("CT22") = TextBox12 / 100 'm
End If

If ComboBox9 = "Direto" And ComboBox10 =
"Indireto" Then
Plan2.Range("CT18") = TextBox9 / 100 'm
Plan2.Range("CT19") = TextBox10 / 100 'm
Plan2.Range("CT21") = 0
Plan2.Range("CT22") = 0
End If

If ComboBox9 = "Indireto" And ComboBox10 =
"Direto" Then
Plan2.Range("CT18") = 0
Plan2.Range("CT19") = 0
Plan2.Range("CT21") = TextBox11 / 100 'm
Plan2.Range("CT22") = TextBox12 / 100 'm
End If

If ComboBox9 = "Indireto" And ComboBox10 =
"Indireto" Then
Plan2.Range("CT18") = 0
Plan2.Range("CT19") = 0
Plan2.Range("CT21") = 0
Plan2.Range("CT22") = 0
End If
'Preenche início da 1ªfaixa e final da 3ªfaixa
xi1 = 0 'cm
Label130 = xi1 'cm
Plan7.Range("B23") = xi1 'cm
xf3 = 100 * L 'cm
Label133 = xf3 'cm
Plan7.Range("B44") = xf3 'cm
End Sub

Private Sub CommandButton1_Click()
If Plan2.Range("CY7") = "MOD1" Then
teta = 3.14159265359 / 4
End If
If Plan2.Range("CY7") = "MOD2" Then
teta = Plan2.Range("CY10")
End If
Set Planilha6 = Sheets("PARCIAL")
xi1 = Plan7.Range("B23")
xf1 = TextBox4
Plan7.Range("B24") = xf1
xi2 = xf1
Label131 = xi2
Plan7.Range("B33") = xi2
comprFaixa1 = xf1 - xi1
Label40 = comprFaixa1
Plan7.Range("B22") = comprFaixa1
'Calcula Cortante Máximo na 1ªfaixa
Call
Functions.Cortante_Biapoitada_Apoio1_Reducao(0)
Vsdmax1 = Plan2.Range("CT11")
Label121 = Vsdmax1
Plan7.Range("B25") = Vsdmax1
'Calcula área de aço na faixa 1
Vc0 = Plan2.Range("CT8")
VRd2 = Plan7.Range("B17")
If Plan2.Range("CY7") = "MOD1" Then
Vc = Plan2.Range("CT6") 'kN
End If
If Plan2.Range("CY7") = "MOD2" And Vc0 >=
Vsdmax1 Then
Vc = Vc0
End If
If Plan2.Range("CY7") = "MOD2" And Vc0 <
Vsdmax1 Then
Vc = Vc0 * (VRd2 - Vsdmax1) / (VRd2 - Vc0)
End If
d = Planilha6.Range("B14") 'cm
If ComboBox8 = "CA50" Then
fywk = 50 'kN/cm²

```

```

Else
    fywk = 50 'kN/cm²
End If
Asw1 = (VSdmax1 - Vc) / (0.9 * (d / 100) * (fywk /
1.15) * (1 / Tan(teta)))
Plan2.Range("CT24") = Asw1
Label47 = Asw1
Plan7.Range("B26") = Asw1
'*****
End Sub

Private Sub CommandButton5_Click()
If Plan2.Range("CY7") = "MOD1" Then
    teta = 3.14159265359 / 4
End If
If Plan2.Range("CY7") = "MOD2" Then
    teta = Plan2.Range("CY10")
End If
    Set Planilha6 = Sheets("PARCIAL")
xi2 = Plan7.Range("B33")
xf2 = TextBox8
Plan7.Range("B34") = xf2
xi3 = xf2
Plan7.Range("B43") = xi3
Label132 = xi3
xf3 = Plan7.Range("B44")
comprFaixa2 = xf2 - xi2
Label59 = comprFaixa2
Plan7.Range("B32") = comprFaixa2
comprFaixa3 = xf3 - xi3
Label78 = comprFaixa3
Plan7.Range("B42") = comprFaixa3
'Preciso localizar os cortantes nos pontos
importantes para calcular Vsdmax2 e 3
Call Functions.Cortante_xf1_xi2(xi2 / 100)
Vsd12 = Plan2.Range("CU11")
Call Functions.Cortante_xf2_xi3(xf2 / 100)
Vsd23 = Plan2.Range("CV11")
Vsdmax2 = Vsd12
    If Vsd23 > Vsdmax2 Then
        Vsdmax2 = Vsd23
    End If
Label124 = Vsdmax2
Plan7.Range("B35") = Vsdmax2
'*****
'Calcula área de aço na faixa 2
Vc0 = Plan2.Range("CT8")
VRd2 = Plan7.Range("B17")
If Plan2.Range("CY7") = "MOD1" Then
    Vc2 = Plan2.Range("CT6") 'kN
End If
If Plan2.Range("CY7") = "MOD2" And Vc0 >=
Vsdmax2 Then
    Vc2 = Vc0
End If
If Plan2.Range("CY7") = "MOD2" And Vc0 <
Vsdmax2 Then
    Vc2 = Vc0 * (VRd2 - Vsdmax2) / (VRd2 - Vc0)
End If
d = Planilha6.Range("B14") 'cm

```

```

If ComboBox8 = "CA50" Then
    fywk = 50 'kN/cm²
Else
    fywk = 50 'kN/cm²
End If
Asw2 = (VSdmax2 - Vc2) / (0.9 * (d / 100) * (fywk /
1.15) * (1 / Tan(teta)))
Plan2.Range("CT25") = Asw2
Label66 = Asw2
Plan7.Range("B36") = Asw2
'*****
L = Plan3.Range("B12") 'm
Call
Functions.Cortante_Biapoiaada_Apoio2_Reducão(L
)
Vsdmax3 = Plan2.Range("CW11")
Label127 = Vsdmax3
Plan7.Range("B45") = Vsdmax3
'*****
'Calcula área de aço na faixa 3
If Plan2.Range("CY7") = "MOD1" Then
    Vc3 = Plan2.Range("CT6") 'kN
End If
If Plan2.Range("CY7") = "MOD2" And Vc0 >=
Vsdmax3 Then
    Vc3 = Vc0
End If
If Plan2.Range("CY7") = "MOD2" And Vc0 <
Vsdmax3 Then
    Vc3 = Vc0 * (VRd2 - Vsdmax3) / (VRd2 - Vc0)
End If
Asw3 = (VSdmax3 - Vc3) / (0.9 * (d / 100) * (fywk /
1.15) * (1 / Tan(teta)))
Plan2.Range("CT26") = Asw3
Label85 = Asw3
Plan7.Range("B46") = Asw3
'*****
End Sub

Private Sub CommandButton2_Click()
Aswmin = Plan2.Range("CT27")
Asw1 = Plan2.Range("CT24")
Asw = Aswmin
If Asw1 > Asw Then
    Asw = Asw1
End If
NR1 = ComboBox1
Plan7.Range("B27") = NR1
Bitola1 = ComboBox2 / 10 'cm
Plan7.Range("B28") = ComboBox2 'mm
espacamento1 = Int(100 / (Asw / (NR1 * 3.14 / 4 *
Bitola1 ^ 2)))
faixa1 = Plan7.Range("B22")
quantidade1 = Int(faixa1 / espacamento1) + 1
Label91 = quantidade1 & " " & ChrW(632) & " " &
Bitola1 * 10 & " c / " & espacamento1 & " (" &
NR1 & " R)"
Plan7.Range("B29") = Label91
AswAdot1 = NR1 * quantidade1 * 3.14 / 4 *
(Bitola1 ^ 2) / (faixa1 / 100)

```



```

Label50 = Format(AswAdot1, "0.00")
Plan7.Range("B30") = Label50
End Sub

Private Sub CommandButton3_Click()
Aswmin = Plan2.Range("CT27")
Asw2 = Plan2.Range("CT25")
Asw = Aswmin
If Asw2 > Asw Then
    Asw = Asw2
End If
NR2 = ComboBox3
Plan7.Range("B37") = NR2
Bitola2 = ComboBox4 / 10 'cm
Plan7.Range("B38") = ComboBox4 'mm
espacamento2 = Int(100 / (Asw / (NR2 * 3.14 / 4 *
Bitola2 ^ 2)))
faixa2 = Plan7.Range("B32")
quantidade2 = Int(faixa2 / espacamento2) + 1
Label92 = quantidade2 & " " & ChrW(632) & " " &
Bitola2 * 10 & " c/ " & espacamento2 & " (" &
NR2 & " R)"
Plan7.Range("B39") = Label92
AswAdot2 = NR2 * quantidade2 * 3.14 / 4 *
(Bitola2 ^ 2) / (faixa2 / 100)
Label69 = Format(AswAdot2, "0.00")
Plan7.Range("B40") = Label69
End Sub

```

```

Private Sub CommandButton4_Click()
Aswmin = Plan2.Range("CT27")
Asw3 = Plan2.Range("CT26")
Asw = Aswmin
If Asw3 > Asw Then
    Asw = Asw3
End If
NR3 = ComboBox5
Plan7.Range("B47") = NR3
Bitola3 = ComboBox6 / 10 'cm
Plan7.Range("B48") = ComboBox6 'mm
espacamento3 = Int(100 / (Asw / (NR3 * 3.14 / 4 *
Bitola3 ^ 2)))
faixa3 = Plan7.Range("B42")
quantidade3 = Int(faixa3 / espacamento3) + 1
Label94 = quantidade3 & " " & ChrW(632) & " " &
Bitola3 * 10 & " c/ " & espacamento3 & " (" &
NR3 & " R)"
Plan7.Range("B49") = Label94
AswAdot3 = NR3 * quantidade3 * 3.14 / 4 *
(Bitola3 ^ 2) / (faixa3 / 100)
Label88 = Format(AswAdot3, "0.00")
Plan7.Range("B50") = Label88
End Sub

```

```
Private Sub OK_Click()
```

```

Unload Me
End Sub

```

Data 9

```

Private Sub UserForm_Initialize()
Set Planilha4 = Sheets("ELS")
Set Planilha6 = Sheets("PARCIAL")
Plan2.Range("DL31") = " "
.....
Nsec = Plan3.Range("C13")
L = Plan3.Range("B12") 'm
'Utiliza mesma seção analisada para fissuração,
ou seja, seção crítica para Combinação Frequente
SecCrit = Plan2.Range("DE1")
xcritico = Plan2.Range("DE2") 'm, posição da
seção crítica
MCFcritico = Plan2.Range("DE3") 'kN.m
Label19 = MCFcritico
Ascritico = Plan2.Range("DE4") 'cm2
d = Plan2.Range("DE5") 'cm
Plan2.Range("DK14") = d
Ap = Plan2.Range("DE7") 'cm2
cg = 100 * Plan3.Range("D8") 'cm
h = 100 * Plan3.Range("B6") 'cm
dp = h - cg + 100 * Planilha4.Range("B" & (24 +
14 * SecCrit)) 'cm
yct = h - cg 'cm
Plan2.Range("DN12") = yct
Label11 = "Seção crítica: " & SecCrit
Label3 = xcritico
.....
FlechaADM = 1000 * L / 250 'mm
Plan2.Range("DE21") = FlechaADM
Label33 = Format(FlechaADM, "0.0")
.....
PinfSecao = Planilha4.Range("B" & (30 + 14 *
SecCrit))
excp = 100 * Planilha4.Range("B" & (24 + 14 *
SecCrit)) 'cm
Winf = Plan3.Range("E10") 'm3
Ac = 10000 * Plan3.Range("B7") 'cm2
fck = Plan3.Range("B16") 'MPa
fctm = 0.3 * fck ^ (2 / 3) * 1000 'kN/m2
fator = Plan2.Range("O17")
aderencia = Plan2.Range("CA2")
fctf = 0.7 * fator * fctm 'kN/m2
Plan2.Range("DE13") = fctf 'kN/m2
Label16 = fctf 'kN/m2
'Cálculo do momento de descompressão
M0 = PinfSecao * (excp / 100 + Winf / (Ac /
10000))
Plan2.Range("DE23") = M0
Mr = M0 + fctf * Winf 'kN.m
Plan2.Range("DE14") = Mr 'kN.m
Call Functions.MomentoCQP(xcritico)
Mcqp = SomaMomentoCQP 'kN.m
Plan2.Range("DE15") = Mcqp 'kN.m
Label6 = Mcqp 'kN.m
.....

```

```

'Calcula tensão de tração e posição da linha neutra
para a combinação CQP

```

```

.....
'Dados para Cálculo do Epré-alongamento
Ep = Plan3.Range("E20") / 10 'kN/cm²
Plan2.Range("DK18") = Ep
Ecs = Plan3.Range("E19") / 10 'kN/cm²
Esi = Plan2.Range("CA20") / 10 'kN/cm²
  alphap = Ep / Ecs
  EpSobreEs = Ep / Esi
  Plan2.Range("DN11") = EpSobreEs
  Plan2.Range("DK16") = Ap 'cm²
  If fator = "1,5" Then
    bc = Plan2.Range("CA3")
  Else
    bc = Plan2.Range("CA4")
  End If
  Plan2.Range("DK12") = bc
  Set Planilha4 = Sheets("ELS")
  Plan2.Range("DK19") = dp
  rop = Ap / (bc * dp)
  Ic = 100000000 * Plan3.Range("D9") 'cm4
  Plan2.Range("DE26") = Ic
  eta = 1 + (excp ^ 2) * Ac / Ic
'Cálculo do Epré-alongamento
If Plan3.Range("C23") = "Pós-Tração" Then
  Eprealongamento = ((PinfSecao / (Ep * Ap)) * (1
+ alphap * rop * eta))
  Else
  Eprealongamento = (PinfSecao / (Ep * Ap))
End If
Plan2.Range("DK17") = Eprealongamento

'Cálculo da força de protensão devida ao pré-
alongamento
  Ninfpref = Ap * Eprealongamento * Ep
  Plan2.Range("DK11") = Ninfpref

'Cálculo do deltaM
  deltaM = (Plan2.Range("DE15") * 100 - Ninfpref *
excp) 'kN.cm
  Plan2.Range("DK9") = deltaM

'Cálculo de sigmas
'xparcial é uma incógnita a ser encontrada, inicia
com valor de H
Sheets("BIBLIOTECA").Select
Plan2.Range("DK10") = 0
Plan2.Range("DK23") = 0

xparcial = 0.2 * h
Plan2.Range("DK10") = xparcial
alphaE = Esi / Ecs
Plan2.Range("DK13") = alphaE
Asparcial = Ascritico
Plan2.Range("DK15") = Asparcial
dsLinha = Plan2.Range("CA7")
Plan2.Range("DN14") = dsLinha
AsLinha = Plan2.Range("CA8")
Plan2.Range("DN15") = AsLinha
.....
'Atingir meta (Ret) ou (T com LN na mesa)
.....
If aderencia = 0 Then
  Plan2.Range("DK22").Select
  Plan2.Range("DK22").GoalSeek Goal:=0,
  ChangingCell:=Plan2.Range("DK10")
End If

If aderencia = 1 Then
  Plan2.Range("DN22").Select
  Plan2.Range("DN22").GoalSeek Goal:=0,
  ChangingCell:=Plan2.Range("DK10")
End If
.....
hf = Plan2.Range("CA5")
Plan2.Range("DK26") = hf
xparcialT = 0.2 * h
Plan2.Range("DK23") = xparcialT
bw = Plan2.Range("CA3")
Plan2.Range("DK24") = bw
bf = Plan2.Range("CA4")
Plan2.Range("DK25") = bf
.....
'Atingir meta T com LN na alma
.....
If aderencia = 0 Then
  Plan2.Range("DK30").Select
  Plan2.Range("DK30").GoalSeek Goal:=0,
  ChangingCell:=Plan2.Range("DK23")
End If

If aderencia = 1 Then
  Plan2.Range("DN30").Select
  Plan2.Range("DN30").GoalSeek Goal:=0,
  ChangingCell:=Plan2.Range("DK23")
End If

.....
'Testa qual Xparcial e sigmaS usar'
.....
xparcialteste = Plan2.Range("DK10")
xparcialverdadeiro = 0
sigmaSverdadeira = 0

If aderencia = 0 Then
  If fator <> 1.5 And xparcialteste > hf Then
    xparcialverdadeiro = Plan2.Range("DK23")
    sigmaSverdadeira = Plan2.Range("DK28")
    Plan2.Range("DK31") = xparcialverdadeiro
    Plan2.Range("DL31") = "LN na alma"
    Plan2.Range("DK32") = sigmaSverdadeira
  Else
    xparcialverdadeiro = Plan2.Range("DK10")
    sigmaSverdadeira = Plan2.Range("DK20")
    Plan2.Range("DK31") = xparcialverdadeiro
    Plan2.Range("DL31") = " "
    Plan2.Range("DK32") = sigmaSverdadeira
  End If
  If fator <> 1.5 And xparcialteste <= hf Then
    Plan2.Range("DL31") = "LN na mesa"
  End If
End If

```

```

End If
End If

If aderencia = 1 Then
  If fator <> 1.5 And xparcialteste > hf Then
    xparcialverdadeiro = Plan2.Range("DK23")
    sigmaSverdadeira = Plan2.Range("DN28")
    Plan2.Range("DK31") = xparcialverdadeiro
    Plan2.Range("DL31") = "LN na alma"
    Plan2.Range("DK32") = sigmaSverdadeira
  Else
    xparcialverdadeiro = Plan2.Range("DK10")
    sigmaSverdadeira = Plan2.Range("DN20")
    Plan2.Range("DK31") = xparcialverdadeiro
    Plan2.Range("DL31") = " "
    Plan2.Range("DK32") = sigmaSverdadeira
  End If
  If fator <> 1.5 And xparcialteste <= hf Then
    Plan2.Range("DL31") = "LN na mesa"
  End If
End If

Plan2.Range("DE6") = xparcialverdadeiro 'cm
xe2 = Plan2.Range("DE6")
.....
  SIGMAinfCF = MCFcritico / Winf - PinfSecao /
(Ac / 10000) - (PinfSecao * excp / 100) / Winf
  Plan2.Range("DE12") = SIGMAinfCF
  Label13 = SIGMAinfCF
.....
  Call Functions.FlechaElasticaG(xcritico)
  FlechaElastG = 1000 * SomaFlechaElasticaG
'mm
  Plan2.Range("DE16") = FlechaElastG

  Call Functions.FlechaElasticaVarCQP(xcritico)
  FlechaElastQ = 1000 * SomaFlechaElasticaQ
'mm
  Plan2.Range("DE25") = FlechaElastQ
.....
  N = Esi / Ecs 'adimensional
  np = Ep / Ecs 'adimensional
  If fator = 1.5 And aderencia = 1 Then
    I2 = (bw * xe2 ^ 3) / 3 + np * Ap * (dp - xe2) ^
2 + N * Ascritico * (d - xe2) ^ 2 + (N - 1) *
AsLinha * (xe2 - dsLinha) ^ 2 'cm4
  End If
  If fator = 1.5 And aderencia = 0 Then
    I2 = (bw * xe2 ^ 3) / 3 + N * Ascritico * (d -
xe2) ^ 2 + (N - 1) * AsLinha * (xe2 - dsLinha) ^ 2
'cm4
  End If
  If fator = 1.2 And xe2 > hf And aderencia = 1
Then
    I2 = ((bf - bw) * hf ^ 3) / 12 + (bf - bw) * hf *
(xe2 - hf / 2) ^ 2 + (bw * xe2 ^ 3) / 3 + N *
Ascritico * (d - xe2) ^ 2 + (N - 1) * AsLinha * (xe2
- dsLinha) ^ 2 'cm4
  End If
  If fator = 1.2 And xe2 <= hf And aderencia = 1
Then
    I2 = (bf * xe2 ^ 3) / 3 + np * Ap * (dp - xe2) ^ 2
+ N * Ascritico * (d - xe2) ^ 2 + (N - 1) * AsLinha
* (xe2 - dsLinha) ^ 2 'cm4
  End If
  If fator = 1.2 And xe2 <= hf And aderencia = 0
Then
    I2 = (bf * xe2 ^ 3) / 3 + N * Ascritico * (d -
xe2) ^ 2 + (N - 1) * AsLinha * (xe2 - dsLinha) ^ 2
'cm4
  End If
  Plan2.Range("DE9") = I2
  Ieq = (((Mr - M0) / (Mcqp - M0)) ^ 3) * Ic + (1 -
((Mr - M0) / (Mcqp - M0)) ^ 3) * I2 'cm4
  Plan2.Range("DG10") = Ieq
  If Ieq > Ic Then
    Ieq = Ic
  End If
  Plan2.Range("DE10") = Ieq
.....
  If SIGMAinfCF > fctf Then
    Plan2.Range("DG12") = "SEÇÃO
FISSURADA"
    FlechaImediataG = FlechaElastG * Ic / Ieq
    FlechaImediataQ = FlechaElastQ * Ic / Ieq
  Else
    Plan2.Range("DG12") = "SEÇÃO NÃO
FISSURADA"
    FlechaImediataG = FlechaElastG
    FlechaImediataQ = FlechaElastQ
  End If
  Plan2.Range("DE17") = FlechaImediataG
  Label21 = FlechaImediataG
  Plan2.Range("DE19") = FlechaImediataQ
  Label27 = FlechaImediataQ
  fi = Data5.TextBox7
  Plan2.Range("DE11") = fi
  FlechaDiferidaG = (1 + fi) * FlechaImediataG
  Plan2.Range("DE18") = FlechaDiferidaG
  Label24 = FlechaDiferidaG
  FlechaFinal = FlechaDiferidaG +
FlechaImediataQ
  Plan2.Range("DE20") = FlechaFinal
  Label30 = FlechaFinal
  If FlechaFinal > FlechaADM Then
    Verificacao = "Flecha maior que a
admissível!!!"
  Else
    Verificacao = "Flecha OK"
  End If
  Plan2.Range("DE22") = Verificacao
  Label36 = Verificacao

```

```

.....
GerarPlanFLECHA ("FLECHAS")
  ActiveSheet.Range("A1:G4").Select
  With Selection
    .HorizontalAlignment = xlCenter
    .VerticalAlignment = xlCenter
    .MergeCells = True
  End With
  With Selection.Font
    .Size = 24
    .Bold = True
  End With
  ActiveSheet.Range("A1") = "VERIFICAÇÃO
DAS FLECHAS"
  ActiveSheet.Range("A5:G5").Merge
  ActiveSheet.Range("A5:G5").Interior.Color =
RGB(220, 230, 241)
  ActiveSheet.Range("A5") = "FLECHA
MÁXIMA"
  ActiveSheet.Range("A6") = "Seção:"
  ActiveSheet.Range("B6") = SecCrit
  ActiveSheet.Range("A7") = "x:"
  ActiveSheet.Range("B7") = xcritico
  ActiveSheet.Range("C7") = "m"
  ActiveSheet.Range("A8") = "Flecha imediata
(g):"
  ActiveSheet.Range("B8") = FlechaImediataG
  ActiveSheet.Range("C8") = "mm"
  ActiveSheet.Range("A9") = "Flecha diferida
(g):"
  ActiveSheet.Range("B9") = FlechaDiferidaG
  ActiveSheet.Range("C9") = "mm"
  ActiveSheet.Range("A10") = "Flecha (q):"
  ActiveSheet.Range("B10") = FlechaImediataQ
  ActiveSheet.Range("C10") = "mm"
  ActiveSheet.Range("A11") = "Flecha final:"
  ActiveSheet.Range("B11") = FlechaFinal
  ActiveSheet.Range("C11") = "mm"
  ActiveSheet.Range("A12") = "Flecha
admissível:"
  ActiveSheet.Range("B12") = FlechaADM
  ActiveSheet.Range("C12") = "mm"
  ActiveSheet.Range("A13") = "Verificação:"
  ActiveSheet.Range("B13") = Verificacao
End Sub

Private Sub CommandButton1_Click()

Unload Me
End Sub

```

Functions

```

Public TENSAO() 'LINHAS:cargas COLUNAS:0-
INFERIOR 1-SUPERIOR

Public MOMENTOS()

Public CABOS() 'CABOS(cabo,seção)=e

```

```

Public EQUIVALENTE()
'EQUIVALENTE(seção,0 (e) ou 1 (alfa))

Public ATRITO() 'ATRITO(seção) = força de
protensão após perda por atrito

Public ANCORAGEM() 'ANCORAGEM(seção) =
força de protensão após perda por deformação da
ancoragem

Public ENCURTAMENTO()
'ENCURTAMENTO(seção) = força de protensão
após perda por encurtamento elástico

Public PROGRESSIVAS()
'PROGRESSIVAS(seção) = força de protensão
após perdas progressivas

Public MomentoCfTotal

Public SomaMomentoCQP

Public SomaFlechaElasticaG

Public SomaFlechaElasticaQ

Function Proteger()
  Plan1.Protect senha
  Plan2.Protect senha
  Plan3.Protect senha
End Function

Function Desproteger()
  Plan1.Unprotect senha
  Plan2.Unprotect senha
  Plan3.Unprotect senha
End Function

Function ExisteArquivo(Arq) As Boolean
Dim fso As FileSystemObject
Dim fld As Folder
Dim fil As File
Set fso = New FileSystemObject
Set fld = fso.GetFolder(ThisWorkbook.Path)
For Each fil In fld.Files
  If UCase(fil.Name) = UCase(Arq) Then
    ExisteArquivo = True
    Exit Function
  Else
    ExisteArquivo = False
  End If
Next fil
End Function

Function Aberto(wkb) As Boolean
Dim T As Excel.Workbook
Err.Clear
On Error Resume Next
Set T = Application.Workbooks(wkb)
Aberto = Not T Is Nothing

```

```

Err.Clear
On Error GoTo 0
End Function

Function Ultima(k)
    Ultima = 1
    For i = 1 To k
        j = ActiveSheet.Cells(Rows.Count,
i).End(xlUp).Row
        If j > Ultima Then Ultima = j
    Next i
End Function

```

```

Function Ultima_A(i As Integer, f As Integer)
    Ultima_A = 1
    x = i
    Do While x <= f
        j = Plan1.Cells(Rows.Count,
x).End(xlUp).Row
        If j > Ultima_A Then Ultima_A = j
        x = x + 1
    Loop
End Function

```

```

Function Ultima_B(i As Integer, f As Integer)
    Ultima_B = 1
    x = i
    Do While x <= f
        j = Plan2.Cells(Rows.Count,
x).End(xlUp).Row
        If j > Ultima_B Then Ultima_B = j
        x = x + 1
    Loop
End Function

```

```

Function carac_especial(ByVal texto As String) As Boolean
    If InStr(texto, "/") > 0 Or InStr(texto, "\") > 0 Or
InStr(texto, ":") > 0 Or InStr(texto, "*") > 0 Or
InStr(texto, "?") > 0 Or InStr(texto, "''") > 0 Or
InStr(texto, "<") > 0 Or InStr(texto, ">") > 0 Or
InStr(texto, "|") > 0 Then
        carac_especial = True
    Else
        carac_especial = False
    End If
End Function

```

```

Function SegundoGrau(a, b, c) 'Deve ser usada
apenas na perda de carga por acomodação da
ancoragem
    d = (b ^ 2) - (4 * a * c)
    raiz1 = 0
    If d > 0 Then
        raiz1 = (-b + Sqr(d)) / (2 * a)
        'raiz2 = (-b - Sqr(d)) / (2 * a)
    Else
        If d = 0 Then
            raiz1 = -b / (2 * a)
            'raiz2 = raiz1

```

```

Else
    'raiz imaginária, não é necessário
End If
End If
If raiz1 > 0 Then
    SegundoGrau = raiz1
Else
    SegundoGrau = 0
End If
End Function

```

```

Function PropGeo()
    ult = Ultima_B(1, 1)
    For i = 2 To ult
        If Plan2.Range("E" & i) = "Ret" Then
            Plan2.Range("F" & i) = Plan2.Range("B" &
i) * Plan2.Range("C" & i)
        Else
            Plan2.Range("F" & i) = (Plan2.Range("B"
& i) * Plan2.Range("C" & i)) / 2
        End If
    Next i
    For i = 2 To ult
        Plan2.Range("G" & i) = Plan2.Range("F" & i)
* Plan2.Range("D" & i)
    Next i
    For i = 2 To ult
        Plan2.Range("H" & i) = Plan2.Range("G" & i)
* Plan2.Range("D" & i)
    Next i
    For i = 2 To ult
        If Plan2.Range("E" & i) = "Ret" Then
            Plan2.Range("I" & i) = (Plan2.Range("B" &
i) * (Plan2.Range("C" & i)) ^ 3) / 12
        Else
            Plan2.Range("I" & i) = (Plan2.Range("B" &
i) * (Plan2.Range("C" & i)) ^ 3) / 36
        End If
    Next i

    If ult > 1 Then
        On Error GoTo ERRO
        h = Plan3.Range("B6")
        Area =
Application.WorksheetFunction.Sum(Plan2.Range(
"F2:F" & ult))
        cg =
(Application.WorksheetFunction.Sum(Plan2.Range
("G2:G" & ult))) / Area
        ig =
Application.WorksheetFunction.Sum(Plan2.Range(
"H2:H" & ult)) +
Application.WorksheetFunction.Sum(Plan2.Range(
"I2:I" & ult)) - (Area * (cg ^ 2))
        Wi = ig / cg
        Ws = -(ig / (h - cg))
        On Error GoTo 0
        Plan3.Range("B7") = Area
        Plan3.Range("D8") = cg
        Plan3.Range("D9") = ig

```

```

Plan3.Range("E10") = Wi
Plan3.Range("E11") = Ws

Plan3.Range("A5:K14").Copy
Destination:=Plan1.Range("A5")
'Plan1.Columns("B:D").EntireColumn.AutoFit
End If
Exit Function
ERRO:
MsgBox "Dados da seção inválidos. Divisão por
zero.", vbCritical, "Dados Inválidos"
End Function

Function ContaCarac(palavra As Variant, carac As
String) As Integer
For i = 1 To Len(palavra)
If Mid(palavra, i, 1) = carac Then
Count = Count + 1
End If
Next i
ContaCarac = Count
End Function

Function ArredondarParaCima(num As Double) As
Double
If num - Int(num) > 0 Then
ArredondarParaCima = Int(num) + 1
Else
ArredondarParaCima = num
End If
End Function

Function Comb_Frequente_Inferior()
ult = Ultima_B(18, 18)
CF = TENSAO(0, 0)
If ult > 1 Then
For linha = 2 To ult
categoria = Plan2.Range("W" & linha)
If categoria = "Perm" Then
CF = CF + TENSAO(linha - 1, 0)
End If
Next linha
CFant = CF
For k = 2 To ult
categoria = Plan2.Range("W" & k)
If categoria = "Var" Then
Psi1 = Plan2.Range("Y" & k)
Psi2 = Plan2.Range("Z" & k)
CFprox = CF + (Psi1 * TENSAO(k - 1,
0))
For j = 2 To ult
categ = Plan2.Range("W" & j)
If categ = "Var" And j <> k Then
CFprox = CFprox + (Psi2 *
TENSAO(j - 1, 0))
End If
Next j
If Abs(CFprox) > Abs(CFant) Then
CFant = CFprox
End If
End If
Next k
CF = CFant
Comb_Frequente_Superior = CF
End Function

Function Comb_Frequente_Superior()
ult = Ultima_B(18, 18)
CFS = TENSAO(0, 1)
If ult > 1 Then
For linha = 2 To ult
categoria = Plan2.Range("W" & linha)
If categoria = "Perm" Then
CFS = CFS + TENSAO(linha - 1, 1)
End If
Next linha
CFSant = CFS
For k = 2 To ult
categoria = Plan2.Range("W" & k)
If categoria = "Var" Then
Psi1 = Plan2.Range("Y" & k)
Psi2 = Plan2.Range("Z" & k)
CFSprox = CFS + (Psi1 * TENSAO(k -
1, 1))
For j = 2 To ult
categ = Plan2.Range("W" & j)
If categ = "Var" And j <> k Then
CFSprox = CFSprox + (Psi2 *
TENSAO(j - 1, 1))
End If
Next j
If Abs(CFSprox) > Abs(CFSant) Then
CFSant = CFSprox
End If
End If
Next k
CFS = CFSant
Comb_Frequente_Superior = CFS
End Function

Function Comb_QuasePermanente_Inferior()
ult = Ultima_B(18, 18)
CQP = TENSAO(0, 0)
If ult > 1 Then
For linha = 2 To ult
categoria = Plan2.Range("W" & linha)
If categoria = "Perm" Then
CQP = CQP + TENSAO(linha - 1, 0)
End If
Next linha
For k = 2 To ult
categoria = Plan2.Range("W" & k)
If categoria = "Var" Then
Psi2 = Plan2.Range("Z" & k)
CQP = CQP + (Psi2 * TENSAO(k - 1, 0))
End If
Next k
End If
End Function

```

```

    Comb_QuasePermanente_Inferior = CQP
End Function

Function Comb_Rara_Inferior()
    ult = Ultima_B(18, 18)
    CR = TENSAO(0, 0)
    If ult > 1 Then
        For linha = 2 To ult
            categoria = Plan2.Range("W" & linha)
            If categoria = "Perm" Then
                CR = CR + TENSAO(linha - 1, 0)
            End If
        Next linha
        CRant = CR
        For k = 2 To ult
            categoria = Plan2.Range("W" & k)
            If categoria = "Var" Then
                Psi1 = Plan2.Range("Y" & k)
                CRprox = CR + TENSAO(k - 1, 0)
                For j = 2 To ult
                    categ = Plan2.Range("W" & j)
                    If categ = "Var" And j <> k Then
                        CRprox = CRprox + (Psi1 *
TENSAO(j - 1, 0))
                    End If
                Next j
                If Abs(CRprox) > Abs(CRant) Then
                    CRant = CRprox
                End If
            End If
        Next k
        CR = CRant
    End If
    Comb_Rara_Inferior = CR
End Function

Function Comb_Rara_Superior()
    ult = Ultima_B(18, 18)
    CRS = TENSAO(0, 1)
    If ult > 1 Then
        For linha = 2 To ult
            categoria = Plan2.Range("W" & linha)
            If categoria = "Perm" Then
                CRS = CRS + TENSAO(linha - 1, 1)
            End If
        Next linha
        CRSant = CRS
        For k = 2 To ult
            categoria = Plan2.Range("W" & k)
            If categoria = "Var" Then
                Psi1 = Plan2.Range("Y" & k)
                CRSprox = CRS + TENSAO(k - 1, 1)
                For j = 2 To ult
                    categ = Plan2.Range("W" & j)
                    If categ = "Var" And j <> k Then
                        CRSprox = CRSprox + (Psi1 *
TENSAO(j - 1, 1))
                    End If
                Next j
                If Abs(CRSprox) > Abs(CRSant) Then

```

```

                CRSant = CRSprox
            End If
        Next k
        CRS = CRSant
    End If
    Comb_Rara_Superior = CRS
End Function

Function Comb_ELU()
    ult = Ultima_B(18, 18)
    MSd = MOMENTOS(0)
    If ult > 1 Then
        For linha = 2 To ult
            categoria = Plan2.Range("W" & linha)
            If categoria = "Perm" Then
                MSd = MSd + MOMENTOS(linha - 1)
            End If
        Next linha
        Msdant = MSd
        For k = 2 To ult
            categoria = Plan2.Range("W" & k)
            If categoria = "Var" Then
                Psi0 = Plan2.Range("X" & k)
                Msdprox = MSd + MOMENTOS(k - 1)
                For j = 2 To ult
                    categ = Plan2.Range("W" & j)
                    If categ = "Var" And j <> k Then
                        Msdprox = Msdprox + (Psi0 *
MOMENTOS(j - 1))
                    End If
                Next j
                If Abs(Msdprox) > Abs(Msdant) Then
                    Msdant = Msdprox
                End If
            End If
        Next k
        MSd = Msdant
    End If
    Comb_ELU = MSd * 1.4
End Function

Function MatrizTensoes(x)
    ult = Ultima_B(18, 18)
    L = Plan3.Range("B12")
    Wi = Plan3.Range("E10")
    Ws = Plan3.Range("E11")
    ReDim TENSAO(0 To ult - 1, 0 To 1)
    'PESO PRÓPRIO
    cargaPP = Plan3.Range("B7") * 25
    MomentoPP = (cargaPP * x * (L / 2)) - (cargaPP
* (x ^ 2) / 2)
    TENSAO(0, 0) = MomentoPP / Wi
    TENSAO(0, 1) = MomentoPP / Ws
    'CARGAS
    If ult > 1 Then
        For linha = 2 To ult
            Tipo = Plan2.Range("S" & linha)
            valor = Plan2.Range("T" & linha)
            xi = Plan2.Range("U" & linha)

```

```

xf = Plan2.Range("V" & linha)
MOMENTO = 0

If Tipo = "Conc." Then
  If x - xi <= 0 Then
    MOMENTO = (valor * (L - xi) * x) / L
  Else
    MOMENTO = (valor * xi * (L - x)) / L
  End If
End If

If Tipo = "Dist." Then
  If x - xi <= 0 Then MOMENTO = valor *
x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
  If (x - xi > 0) And (x - xf < 0) Then
    MOMENTO = (valor * x * ((xf - xi) - ((xf ^ 2) - (xi
^ 2)) / (2 * L))) - (valor * ((x - xi) ^ 2) / 2)
  If x - xf >= 0 Then MOMENTO = valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L - x)
  End If
  MOMENTOS(linha - 1) = MOMENTO
Next linha
End If
MSolD = 1.4 * MomentoPP
For i = 1 To ult - 1
  MSolD = MSolD + 1.4 * MOMENTOS(i)
Next i
Plan2.Range("CT5") = Format(MSolD, "0.00")
End Function

Function Tracado_Cabos()
  ult = Ultima_B(65, 65)
  Nsec = Plan3.Range("C13")
  L = Plan3.Range("B12")

  If ult > 1 Then
    If
Application.WorksheetFunction.IsEven(Nsec) Then
      num_sec = Nsec / 2
      'meio = 0
    Else
      num_sec = (Nsec + 1) / 2
      'meio = 1
    End If
    ReDim CABOS(1 To (ult - 1), 0 To Nsec)
    For linha = 2 To ult
      Cabo = linha - 1
      ei = Plan2.Range("BN" & linha)
      xm = Plan2.Range("BO" & linha)
      em = Plan2.Range("BP" & linha)
      For i = 1 To num_sec
        x = ((i - 1) * L) / (Nsec - 1)
        If (x - xm) < 0 And xm <> 0 Then e = (((-
em + ei) / (xm ^ 2)) * (x ^ 2)) + ((2 * (em - ei) / xm)
* x) + ei
        If (x - xm) >= 0 Or xm = 0 Then e = em
        CABOS(Cabo, i) = e
        CABOS(Cabo, Nsec + 1 - i) = e
      Next i
      'meio do vão x=L/2
      CABOS(Cabo, 0) = em
    Next linha
  End If
End Function

Function MomentoPermanenteBalanceada()
  'calcula momento permanente a balancear
  momentopermbalanc = 0
  If Plan6.Range("B6") <> 0 Then
    momentopermbalanc = Plan6.Range("B5") *
Plan6.Range("B6") / 100
  End If
  Plan6.Range("B7") = momentopermbalanc
End Function

Function VetorMomentos(x)
  ult = Ultima_B(18, 18)
  L = Plan3.Range("B12")
  ReDim MOMENTOS(0 To ult - 1)
  'PESO PRÓPRIO
  cargaPP = Plan3.Range("B7") * 25
  MomentoPP = (cargaPP * x * (L / 2)) - (cargaPP
* (x ^ 2) / 2)
  MOMENTOS(0) = MomentoPP

  'CARGAS
  If ult > 1 Then
    For linha = 2 To ult
      Tipo = Plan2.Range("S" & linha)
      valor = Plan2.Range("T" & linha)
      xi = Plan2.Range("U" & linha)
      xf = Plan2.Range("V" & linha)

      MOMENTO = 0

      If Tipo = "Conc." Then
        If x - xi <= 0 Then
          MOMENTO = (valor * (L - xi) * x) / L
        Else
          MOMENTO = (valor * xi * (L - x)) / L
        End If
      End If

      If Tipo = "Dist." Then
        If x - xi <= 0 Then MOMENTO = valor *
x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
        If (x - xi > 0) And (x - xf < 0) Then
          MOMENTO = (valor * x * ((xf - xi) - ((xf ^ 2) - (xi
^ 2)) / (2 * L))) - (valor * ((x - xi) ^ 2) / 2)
        If x - xf >= 0 Then MOMENTO = valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L - x)
        End If
        MOMENTOS(linha - 1) = MOMENTO
      Next linha
    End If
  End Function

```



```

'cargas permanentes
'Cada linha tem o momento devido a uma carga
permanente
'Esse vetor é todo para uma seção apenas
'SomaMomentoPerm soma todas as linhas e resulta
no momento total devido as cargas permanentes

```

```

Function SomaMomentoPerm(x)
    ult = Ultima_B(18, 18)
    L = Plan3.Range("B12")
    ReDim MOMENTOSpermanentes(0 To ult - 1)
    'PESO PRÓPRIO
    cargaPP = Plan3.Range("B7") * 25
    MomentoPP = (cargaPP * x * (L / 2)) - (cargaPP
* (x ^ 2) / 2)
    MOMENTOSpermanentes(0) = MomentoPP

    'CARGAS
    If ult > 1 Then
        For linha = 2 To ult
            Tipo = Plan2.Range("S" & linha)
            valor = Plan2.Range("T" & linha)
            xi = Plan2.Range("U" & linha)
            xf = Plan2.Range("V" & linha)
            categoria = Plan2.Range("W" & linha)
            MOMENTO = 0
            If categoria = "Perm" Then
                If Tipo = "Conc." Then
                    If x - xi <= 0 Then
                        MOMENTO = (valor * (L - xi) * x) / L
                    Else
                        MOMENTO = (valor * xi * (L - x)) / L
                    End If
                End If

                If Tipo = "Dist." Then
                    If x - xi <= 0 Then MOMENTO = valor *
x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
                    If (x - xi > 0) And (x - xf < 0) Then
                        MOMENTO = (valor * x * ((xf - xi) - ((xf ^ 2) - (xi
^ 2)) / (2 * L))) - (valor * ((x - xi) ^ 2) / 2)
                    If x - xf >= 0 Then MOMENTO = valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L - x)
                    End If
                End If
            End If
            MOMENTOSpermanentes(linha - 1) =
MOMENTO
        End If
    Next linha
    End If
    SomaMomentoPerm = 0
    For j = 0 To ult - 1
        SomaMomentoPerm = SomaMomentoPerm +
MOMENTOSpermanentes(j)
    Next j
    Plan6.Range("B5") = SomaMomentoPerm
End Function

```

```

'Parcela acidental da Combinação Frequente
'Cada linha tem o momento devido a (uma carga
acidental com psi 1 e as demais com psi2)
'MomentoAcidMax procura o maior momento entre
as parcelas acidentais
'Esse vetor é todo para uma seção apenas

```

```

Function MomentoAcidMax(x)
    ult = Ultima_B(18, 18)
    L = Plan3.Range("B12")
    ReDim MOMENTOSvariaveis(0 To ult - 1)
    MOMENTOSvariaveis(0) = 0

    'CARGAS
    If ult > 1 Then
        For linha = 2 To ult
            Tipo = Plan2.Range("S" & linha)
            valor = Plan2.Range("T" & linha)
            xi = Plan2.Range("U" & linha)
            xf = Plan2.Range("V" & linha)
            categoria = Plan2.Range("W" & linha)

            MOMENTOacid = 0
            If categoria = "Var" Then
                Psi1 = Plan2.Range("Y" & linha)
                If Tipo = "Conc." Then
                    If x - xi <= 0 Then
                        MOMENTO1 = Psi1 * (valor * (L - xi)
* x) / L
                    Else
                        MOMENTO1 = Psi1 * (valor * xi * (L
- x)) / L
                    End If
                End If

                If Tipo = "Dist." Then
                    If x - xi <= 0 Then MOMENTO1 = Psi1
* (valor * x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 *
L)))
                    If (x - xi > 0) And (x - xf < 0) Then
                        MOMENTO1 = Psi1 * (((valor * x * ((xf - xi) - ((xf
^ 2) - (xi ^ 2)) / (2 * L))) - (valor * ((x - xi) ^ 2) /
2))
                    If x - xf >= 0 Then MOMENTO1 = Psi1
* (valor * (((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L - x))
                    End If
                End If
            End If
            MOMENTOSvariaveis(linha - 1) =
MOMENTO1

            For j = 2 To ult
                categ = Plan2.Range("W" & j)
                If categ = "Var" And j <> linha Then
                    Psi2 = Plan2.Range("Z" & j)
                    q2 = Plan2.Range("T" & j)
                    If Tipo = "Conc." Then
                        If x - xi <= 0 Then
                            MOMENTO2 = Psi2 * (q2 * (L -
xi) * x) / L
                        Else

```

```

                MOMENTO2 = Psi2 * (q2 * xi *
(L - x)) / L
                End If
            End If

            If Tipo = "Dist." Then
                If x - xi <= 0 Then MOMENTO2
= Psi2 * (q2 * x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2
* L)))
                If (x - xi > 0) And (x - xf < 0)
Then MOMENTO2 = Psi2 * ((q2 * x * ((xf - xi) -
((xf ^ 2) - (xi ^ 2)) / (2 * L))) - (q2 * ((x - xi) ^ 2) /
2))
                If x - xf >= 0 Then MOMENTO2
= Psi2 * (q2 * (((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L -
x))
            End If
            MOMENTOSvariaveis(linha - 1) =
MOMENTOSvariaveis(linha - 1) + MOMENTO2
        End If
    Next j
End If
Next linha
End If
MomentoAcidMax = MOMENTOSvariaveis(0)
For k = 1 To ult - 1
    If MomentoAcidMax <
MOMENTOSvariaveis(k) Then
        MomentoAcidMax =
MOMENTOSvariaveis(k)
    End If
Next k
End Function

```

.....
'Essa função soma o momento total permanente
com o maior acidental e resulta na combinação
frequente

'Esse vetor é todo para uma seção apenas
.....

Function MomentoCF(x)

```

aux1 = SomaMomentoPerm(x)
aux2 = MomentoAcidMax(x)
MomentoCfTotal = aux1 + aux2
Plan6.Range("B8") = MomentoCfTotal * 100
End Function

```

.....
'COMBINAÇÃO QUASE PERMANENTE
' Cada linha tem o momento devido a uma carga
' Esse vetor é todo para uma seção apenas
' SomaMomentoCQP soma todas as linhas e resulta
no momento total devido a todas as cargas
.....

```

Function MomentoCQP(x)
    ult = Ultima_B(18, 18)
    L = Plan3.Range("B12")

```

```

ReDim MomentosCQP(0 To ult - 1)
' PESO PRÓPRIO
cargaPP = Plan3.Range("B7") * 25
MomentoPP = (cargaPP * x * (L / 2)) - (cargaPP
* (x ^ 2) / 2)
MomentosCQP(0) = MomentoPP

```

' CARGAS

If ult > 1 Then

For linha = 2 To ult

Tipo = Plan2.Range("S" & linha)

valor = Plan2.Range("T" & linha)

xi = Plan2.Range("U" & linha)

xf = Plan2.Range("V" & linha)

categoria = Plan2.Range("W" & linha)

Psi2 = Plan2.Range("Z" & linha)

MOMENTO = 0

If categoria = "Perm" Then

If Tipo = "Conc." Then

If x - xi <= 0 Then

MOMENTO = (valor * (L - xi) * x) / L

Else

MOMENTO = (valor * xi * (L - x)) / L

End If

End If

If Tipo = "Dist." Then

If x - xi <= 0 Then MOMENTO = valor *
x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))

If (x - xi > 0) And (x - xf < 0) Then
MOMENTO = (valor * x * ((xf - xi) - ((xf ^ 2) - (xi
^ 2)) / (2 * L))) - (valor * ((x - xi) ^ 2) / 2)

If x - xf >= 0 Then MOMENTO = valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L - x)

End If

MomentosCQP(linha - 1) = MOMENTO

End If

If categoria = "Var" Then

If Tipo = "Conc." Then

If x - xi <= 0 Then

MOMENTO = Psi2 * (valor * (L - xi)
* x) / L

Else

MOMENTO = Psi2 * (valor * xi * (L -
x)) / L

End If

End If

If Tipo = "Dist." Then

If x - xi <= 0 Then MOMENTO = Psi2 *
(valor * x * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L)))

If (x - xi > 0) And (x - xf < 0) Then
MOMENTO = Psi2 * ((valor * x * ((xf - xi) - ((xf ^
2) - (xi ^ 2)) / (2 * L))) - (valor * ((x - xi) ^ 2) / 2)

If x - xf >= 0 Then MOMENTO = Psi2 *
(valor * (((xf ^ 2) - (xi ^ 2)) / (2 * L)) * (L - x))

End If

MomentosCQP(linha - 1) = MOMENTO

```

    End If
    Next linha
End If
SomaMomentoCQP = 0
For j = 0 To ult - 1
    SomaMomentoCQP = SomaMomentoCQP +
MomentosCQP(j)
    Next j
End Function

'FLECHA ELÁSTICA DEVIDA AS CARGAS
PERMANENTES E ALÍVIO PROTENSÃO,
OBTIDA PELA INTEGRAÇÃO DOS
MOMENTOS
'Cada linha tem a flecha devido a uma carga
'Esse vetor é todo para uma seção apenas
'SomaFlechaElasticaG soma todas as linhas e
resulta na flecha elástica devida a todas as cargas
permanentes mais alívio devido a protensão
Function FlechaElasticaG(x)
    ult = Ultima_B(18, 18)
    L = Plan3.Range("B12") 'm
    fck = Plan3.Range("B16") 'MPa
    e = 1000 * Plan3.Range("E19") 'kN/m2
    Itotal = Plan3.Range("D9") 'm4
    ReDim FlechaElastica(0 To ult)
    'PESO PRÓPRIO
    cargaPP = Plan3.Range("B7") * 25 'kN/m
    FlechaPP = (-1 / (e * Itotal)) * ((L * cargaPP * x
^ 3) / 12 - (cargaPP * x ^ 4) / 24 - ((L ^ 3) *
cargaPP * x) / 24)
    FlechaElastica(0) = FlechaPP 'm

    'CARGAS
    If ult > 1 Then
        For linha = 2 To ult
            Tipo = Plan2.Range("S" & linha)
            valor = Plan2.Range("T" & linha)
            xi = Plan2.Range("U" & linha)
            xf = Plan2.Range("V" & linha)
            categoria = Plan2.Range("W" & linha)

            FLECHA = 0
            If categoria = "Perm" Then
                If Tipo = "Conc." Then
                    If x - xi <= 0 Then
                        FLECHA = (-1 / (e * Itotal)) * ((L *
valor / 6) * (xi - L) * x + ((x ^ 3) * (L * valor -
valor * xi)) / (6 * L))
                    Else
                        FLECHA = (-1 / (e * Itotal)) * ((L *
valor * xi / 6 - L * valor * xi / 2) * x + (valor * xi *
x ^ 2) / 2 - (valor * xi * x ^ 3) / (6 * L))
                    End If
                End If
            End If

            If Tipo = "Dist." Then

```

```

                If x - xi <= 0 Then FLECHA = (-1 / (e *
Itotal)) * ((L * valor / 6) * (L * (xi - xf) + (1 / 2) *
(xf ^ 2 - xi ^ 2)) * x - (x ^ 3) * (valor * xf ^ 2 -
valor * xi ^ 2 - 2 * L * valor * xf + 2 * L * valor *
xi) / (12 * L))
                If (x - xi > 0) And (x - xf < 0) Then
                    FLECHA = (-1 / (e * Itotal)) * (valor * L * ((L ^ 2)
/ 24 - (xf * L) / 6 + (xi ^ 2) / 4 + (xf ^ 2) / 12 - (xi ^
2) / 12) * x - (valor * x ^ 4) / 24 - (valor * (x ^ 2) *
(xi ^ 2) / 4 + ((x ^ 3) * (valor * (xi ^ 2) - valor *
(xf ^ 2) + 2 * L * valor * xf)) / (12 * L))
                If x - xf >= 0 Then FLECHA = (-1 / (e *
Itotal)) * (x * ((L * valor * xi ^ 2) / 4 - (L * valor *
xf ^ 2) / 4 + (L * valor / 12) * (xf ^ 2 - xi ^ 2)) + (x
^ 2) * ((valor * xf ^ 2) / 4 - (valor * xi ^ 2) / 4) - (x
^ 3) * (valor * xf ^ 2 - valor * xi ^ 2) / 12 * L)
                End If
                FlechaElastica(linha - 1) = FLECHA
            End If

            If categoria = "Var" Then
                FlechaElastica(linha - 1) = 0
            End If
        Next linha
    End If

    'PROTENSÃO
    Set Planilha7 = Sheets("CORTANTE")
    qALIV = Planilha7.Range("B6")
    FlechaPROT = (1 / (e * Itotal)) * ((L * qALIV *
x ^ 3) / 12 - (qALIV * x ^ 4) / 24 - ((L ^ 3) *
qALIV * x) / 24)
    FlechaElastica(ult) = FlechaPROT

    SomaFlechaElasticaG = 0
    For j = 0 To ult
        SomaFlechaElasticaG = SomaFlechaElasticaG +
FlechaElastica(j) 'm
    Next j
End Function

'FLECHA ELÁSTICA DEVIDA AS CARGAS
VARIÁVEIS, OBTIDA PELA INTEGRAÇÃO
DOS MOMENTOS NA COMBINAÇÃO QUASE
PERMANENTE
'Cada linha tem a flecha devido a uma carga
'Esse vetor é todo para uma seção apenas
'SomaFlechaElasticaQ soma todas as linhas e
resulta na flecha elástica devida a todas as cargas
variáveis
Function FlechaElasticaVarCQP(x)
    ult = Ultima_B(18, 18)
    L = Plan3.Range("B12") 'm
    fck = Plan3.Range("B16") 'MPa
    e = 1000 * Plan3.Range("E19") 'kN/m2
    Itotal = Plan3.Range("D9") 'm4
    ReDim FlechaElastica(0 To ult)

```

```

FlechaElastica(0) = 0 'm

'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)
    categoria = Plan2.Range("W" & linha)
    Psi2 = Plan2.Range("Z" & linha)

    FLECHA = 0
    If categoria = "Perm" Then
      FlechaElastica(linha - 1) = 0
    End If

    If categoria = "Var" Then
      If Tipo = "Conc." Then
        If x - xi <= 0 Then
          FLECHA = (-Psi2 / (e * Itotal)) * ((L *
valor / 6) * (xi - L) * x + ((x ^ 3) * (L * valor -
valor * xi)) / (6 * L))
        Else
          FLECHA = (-Psi2 / (e * Itotal)) * ((L *
valor * xi / 6 - L * valor * xi / 2) * x + (valor * xi *
x ^ 2) / 2 - (valor * xi * x ^ 3) / (6 * L))
        End If
      End If

      If Tipo = "Dist." Then
        If x - xi <= 0 Then FLECHA = (-Psi2 / (e
* Itotal)) * ((L * valor / 6) * (L * (xi - xf) + (1 / 2) *
(xf ^ 2 - xi ^ 2)) * x - (x ^ 3) * (valor * xf ^ 2 -
valor * xi ^ 2 - 2 * L * valor * xf + 2 * L * valor *
xi) / (12 * L))
        If (x - xi > 0) And (x - xf < 0) Then
          FLECHA = (-Psi2 / (e * Itotal)) * (valor * L * ((L ^
2) / 24 - (xf * L) / 6 + (xi ^ 2) / 4 + (xf ^ 2) / 12 - (xi
^ 2) / 12) * x - (valor * x ^ 4) / 24 - (valor * (x ^ 2)
* (xi ^ 2)) / 4 + ((x ^ 3) * (valor * (xi ^ 2) - valor *
(xf ^ 2) + 2 * L * valor * xf)) / (12 * L))
        If x - xf >= 0 Then FLECHA = (-Psi2 / (e
* Itotal)) * (x * ((L * valor * xi ^ 2) / 4 - (L * valor
* xf ^ 2) / 4 + (L * valor / 12) * (xf ^ 2 - xi ^ 2)) +
(x ^ 2) * ((valor * xf ^ 2) / 4 - (valor * xi ^ 2) / 4) -
(x ^ 3) * (valor * xf ^ 2 - valor * xi ^ 2) / 12 * L)
        End If
      End If
      FlechaElastica(linha - 1) = FLECHA
    End If
  Next linha
End If

FlechaElastica(ult) = 0

SomaFlechaElasticaQ = 0
For j = 0 To ult
  SomaFlechaElasticaQ = SomaFlechaElasticaQ +
FlechaElastica(j) 'm
Next j

```

```

End Function

Function VetorCortantes(x)
  ult = Ultima_B(18, 18)
  L = Plan3.Range("B12")
  ReDim CORTANTES(0 To ult)
  'PESO PRÓPRIO
  cargaPP = Plan3.Range("B7") * 25
  CortantePP = (cargaPP * (L / 2)) - (cargaPP * x)
  CORTANTES(0) = CortantePP
  'CARGAS
  If ult > 1 Then
    For linha = 2 To ult
      Tipo = Plan2.Range("S" & linha)
      valor = Plan2.Range("T" & linha)
      xi = Plan2.Range("U" & linha)
      xf = Plan2.Range("V" & linha)

      CORTANTE = 0

      If Tipo = "Conc." Then
        If x - xi <= 0 Then
          CORTANTE = (valor * (L - xi)) / L
        Else
          CORTANTE = (-valor * xi) / L
        End If
      End If

      If Tipo = "Dist." Then
        If x - xi <= 0 Then CORTANTE = valor
* ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
        If (x - xi > 0) And (x - xf < 0) Then
          CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
        If x - xf >= 0 Then CORTANTE = -valor
* (((xf ^ 2) - (xi ^ 2)) / (2 * L))
        End If
      End If
      CORTANTES(linha - 1) = CORTANTE

      Next linha
    End If
  'PROTENSÃO
  qALIV = Plan7.Range("B6")
  CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
  CORTANTES(ult) = CortantePROT
  Vg_Vq = 0
  For i = 0 To ult - 1
    Vg_Vq = Vg_Vq + CORTANTES(i)
  Next i

  Plan7.Range("B8") = Format(Vg_Vq, "0.00")
  Vpinfinito = CORTANTES(ult)
  Plan7.Range("B9") = Format(Vpinfinito, "0.00")

  VSd = 1.4 * Vg_Vq + 0.9 * Vpinfinito
  Plan7.Range("B10") = Format(VSd, "0.00")
End Function

```

```

Function Cortante_Biapoiaada_Apoio1(x)
ult = Ultima_B(18, 18)
L = Plan3.Range("B12")
ReDim CORTANTES(0 To ult)
'PESO PRÓPRIO
cargaPP = Plan3.Range("B7") * 25
CortantePP = (cargaPP * (L / 2)) - (cargaPP * x)
CORTANTES(0) = CortantePP
'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)

    CORTANTE = 0

    If Tipo = "Conc." Then
      If x - xi <= 0 Then
        CORTANTE = (valor * (L - xi)) / L
      Else
        CORTANTE = (-valor * xi) / L
      End If
    End If

    If Tipo = "Dist." Then
      If x - xi <= 0 Then CORTANTE = valor
      * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
      If (x - xi > 0) And (x - xf < 0) Then
        CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
      If x - xf >= 0 Then CORTANTE = -valor
      * (((xf ^ 2) - (xi ^ 2)) / (2 * L))
      End If
    End If
    CORTANTES(linha - 1) = CORTANTE

    Next linha
  End If
  'PROTENSÃO
  qALIV = Plan7.Range("B6")
  CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
  CORTANTES(ult) = CortantePROT

  Vg_Vq = 0
  For i = 0 To ult - 1
    Vg_Vq = Vg_Vq + CORTANTES(i)
  Next i

  Vpinfinito = CORTANTES(ult)

  VSd1 = 1.4 * Vg_Vq + 0.9 * Vpinfinito
  Plan2.Range("CT2") = VSd1
End Function

Function Cortante_Biapoiaada_Apoio2(x)
ult = Ultima_B(18, 18)
L = Plan3.Range("B12")
ReDim CORTANTES(0 To ult)

```

```

'PESO PRÓPRIO
cargaPP = Plan3.Range("B7") * 25
CortantePP = (cargaPP * (L / 2)) - (cargaPP * x)
CORTANTES(0) = CortantePP
'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)

    CORTANTE = 0

    If Tipo = "Conc." Then
      If x - xi <= 0 Then
        CORTANTE = (valor * (L - xi)) / L
      Else
        CORTANTE = (-valor * xi) / L
      End If
    End If

    If Tipo = "Dist." Then
      If x - xi <= 0 Then CORTANTE = valor
      * ((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
      If (x - xi > 0) And (x - xf < 0) Then
        CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
      If x - xf >= 0 Then CORTANTE = -valor
      * (((xf ^ 2) - (xi ^ 2)) / (2 * L))
      End If
    End If
    CORTANTES(linha - 1) = CORTANTE

    Next linha
  End If
  'PROTENSÃO
  qALIV = Plan7.Range("B6")
  CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
  CORTANTES(ult) = CortantePROT

  Vg_Vq = 0
  For i = 0 To ult - 1
    Vg_Vq = Vg_Vq + CORTANTES(i)
  Next i

  Vpinfinito = CORTANTES(ult)

  VSd2 = 1.4 * Vg_Vq + 0.9 * Vpinfinito
  Plan2.Range("CT3") = VSd2
End Function

Function Cortante_Biapoiaada_Apoio1_Reducão(x)
'Essa função deve ser utilizada para dimensionar a
armação na primeira faixa.
d = Plan6.Range("B14") / 100 'm
Plan2.Range("CT15") = d
L = Plan3.Range("B12") 'm
Plan2.Range("CT16") = L

```

```

tipoApoio = Plan2.Range("CT17")
ax = Plan2.Range("CT18")
cgx = Plan2.Range("CT19")
xRedConc = 2 * d
xRedDist = cgx + d / 2

ult = Ultima_B(18, 18)
ReDim CORTANTES(0 To ult)
'PESO PRÓPRIO
cargaPP = Plan3.Range("B7") * 25
CortantePP = (cargaPP * (L / 2)) - (cargaPP *
x)

CORTANTES(0) = CortantePP
'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)

    CORTANTE = 0

    If Tipo = "Conc." Then
      If tipoApoio = "Direto" And x <=
xRedConc And Plan2.Range("CP19") = "SIM"
Then
        If x - xi <= 0 Then
          CORTANTE = ((x / xRedConc) *
valor * (L - xi)) / L
        Else
          CORTANTE = (-(x / xRedConc) *
valor * xi) / L
        End If
      End If
      If tipoApoio = "Indireto" Or x > xRedConc
Or Plan2.Range("CP19") = "NAO" Then
        If x - xi <= 0 Then
          CORTANTE = (valor * (L - xi)) / L
        Else
          CORTANTE = (-valor * xi) / L
        End If
      End If
    End If

    If Tipo = "Dist." Then
      If tipoApoio = "Direto" And x <=
xRedDist And Plan2.Range("CP19") = "SIM" Then
        If x - xi < 0 Then CORTANTE = valor *
(((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
        If (x - xi >= 0) And (x - xf <= 0) Then
          CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (xRedDist - xi))
        If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
        End If
      End If
      If tipoApoio = "Indireto" Or x > xRedDist
Or Plan2.Range("CP19") = "NAO" Then
        If x - xi < 0 Then CORTANTE = valor *
(((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
        If (x - xi >= 0) And (x - xf <= 0) Then
          CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
        If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
        End If
      End If
    End If
  Next linha
End If
'PROTENSÃO
qALIV = Plan7.Range("B6")
CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
CORTANTES(ult) = CortantePROT

Vg_Vq = 0
For i = 0 To ult - 1
  Vg_Vq = Vg_Vq + CORTANTES(i)
Next i

Vpinfinito = CORTANTES(ult)

VSd1 = 1.4 * Vg_Vq + 0.9 * Vpinfinito

Plan2.Range("CT11") = Abs(VSd1) 'cortante em
xi1
End Function

Function Cortante_Biapoioda_Apoio2_Reducão(x)
'Essa função deve ser utilizada para dimensionar a
armação na terceira faixa.
d = Plan6.Range("B14") / 100 'm
Plan2.Range("CT15") = d
L = Plan3.Range("B12") 'm
Plan2.Range("CT16") = L
tipoApoio = Plan2.Range("CT20")
ax = Plan2.Range("CT21")
cgx = Plan2.Range("CT22")
xRedConc = 2 * d
xRedDist = (cgx + d / 2)

ult = Ultima_B(18, 18)
ReDim CORTANTES(0 To ult)

'PESO PRÓPRIO
cargaPP = Plan3.Range("B7") * 25
CortantePP = (cargaPP * (L / 2)) - (cargaPP *
x)

CORTANTES(0) = CortantePP
'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)

```

```

CORTANTE = 0

If Tipo = "Conc." Then
  If tipoApoio = "Direto" And x >= L -
xRedConc And Plan2.Range("CP19") = "SIM"
  Then
    If x - xi <= 0 Then
      CORTANTE = (((L - x) / xRedConc) *
valor * (L - xi)) / L
    Else
      CORTANTE = (-((L - x) / xRedConc)
* valor * xi) / L
    End If
  End If
  If tipoApoio = "Indireto" Or x < L -
xRedConc Or Plan2.Range("CP19") = "NAO"
  Then
    If x - xi <= 0 Then
      CORTANTE = (valor * (L - xi)) / L
    Else
      CORTANTE = (-valor * xi) / L
    End If
  End If
End If

If Tipo = "Dist." Then
  If tipoApoio = "Direto" And x >= L -
xRedDist And Plan2.Range("CP19") = "SIM" Then
    If x - xi < 0 Then CORTANTE = valor *
((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
    If (x - xi >= 0) And (x - xf <= 0) Then
CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * ((L - xRedDist) - xi))
    If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
    End If
  If tipoApoio = "Indireto" Or x < L -
xRedDist Or Plan2.Range("CP19") = "NAO" Then
    If x - xi < 0 Then CORTANTE = valor *
((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
    If (x - xi >= 0) And (x - xf <= 0) Then
CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
    If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
    End If
  End If
  CORTANTES(linha - 1) = CORTANTE

  Next linha
End If
'PROTENSÃO
qALIV = Plan7.Range("B6")
CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
CORTANTES(ult) = CortantePROT

Vg_Vq = 0
For i = 0 To ult - 1
  Vg_Vq = Vg_Vq + CORTANTES(i)

```

```

Next i

Vpinfinito = CORTANTES(ult)

VSd2 = 1.4 * Vg_Vq + 0.9 * Vpinfinito

Plan2.Range("CW11") = Abs(VSd2) 'cortante
em xf3
End Function

Function Cortante_xf1_xi2(x)
'Essa função deve ser utilizada para dimensionar
calcular Vsd em xf1=xi2.
d = Plan6.Range("B14") / 100 'm
Plan2.Range("CT15") = d
L = Plan3.Range("B12") 'm
Plan2.Range("CT16") = L
tipoApoio = Plan2.Range("CT17")
ax = Plan2.Range("CT18")
cgx = Plan2.Range("CT19")
xRedConc = 2 * d
xRedDist = cgx + d / 2

ult = Ultima_B(18, 18)
ReDim CORTANTES(0 To ult)

'PESO PRÓPRIO
cargaPP = Plan3.Range("B7") * 25
CortantePP = (cargaPP * (L / 2)) - (cargaPP *
x)

CORTANTES(0) = CortantePP
'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)

    CORTANTE = 0

    If Tipo = "Conc." Then
      If tipoApoio = "Direto" And x <=
xRedConc And Plan2.Range("CP19") = "SIM"
      Then
        If x - xi <= 0 Then
          CORTANTE = ((x / xRedConc) *
valor * (L - xi)) / L
        Else
          CORTANTE = -(x / xRedConc) *
valor * xi) / L
        End If
      End If
      If tipoApoio = "Indireto" Or x > xRedConc
Or Plan2.Range("CP19") = "NAO" Then
        If x - xi <= 0 Then
          CORTANTE = (valor * (L - xi)) / L
        Else
          CORTANTE = (-valor * xi) / L

```

```

    End If
  End If
End If

If Tipo = "Dist." Then
  If tipoApoio = "Direto" And x <=
xRedDist And Plan2.Range("CP19") = "SIM" Then
    If x - xi < 0 Then CORTANTE = valor *
((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
    If (x - xi >= 0) And (x - xf <= 0) Then
CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (xRedDist - xi))
    If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
    End If
    If tipoApoio = "Indireto" Or x > xRedDist
Or Plan2.Range("CP19") = "NAO" Then
    If x - xi < 0 Then CORTANTE = valor *
((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
    If (x - xi >= 0) And (x - xf <= 0) Then
CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
    If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
    End If
  End If
  CORTANTES(linha - 1) = CORTANTE

  Next linha
End If
'PROTENSÃO
qALIV = Plan7.Range("B6")
CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
CORTANTES(ult) = CortantePROT

Vg_Vq = 0
For i = 0 To ult - 1
  Vg_Vq = Vg_Vq + CORTANTES(i)
Next i

Vpinfinito = CORTANTES(ult)

Vsd12 = 1.4 * Vg_Vq + 0.9 * Vpinfinito

Plan2.Range("CU11") = Abs(Vsd12) 'cortante
em xfl=xi2
End Function

Function Cortante_xf2_xi3(x)
'Essa função deve ser utilizada para dimensionar
calcular Vsd em xf2=xi3.
  d = Plan6.Range("B14") / 100 'm
  Plan2.Range("CT15") = d
  L = Plan3.Range("B12") 'm
  Plan2.Range("CT16") = L
  tipoApoio = Plan2.Range("CT20")
  ax = Plan2.Range("CT21")
  cgx = Plan2.Range("CT22")
  xRedConc = 2 * d

```

```

  xRedDist = (cgx + d / 2)

ult = Ultima_B(18, 18)
ReDim CORTANTES(0 To ult)

'PESO PRÓPRIO
  cargaPP = Plan3.Range("B7") * 25
  CortantePP = (cargaPP * (L / 2)) - (cargaPP *
x)

CORTANTES(0) = CortantePP
'CARGAS
If ult > 1 Then
  For linha = 2 To ult
    Tipo = Plan2.Range("S" & linha)
    valor = Plan2.Range("T" & linha)
    xi = Plan2.Range("U" & linha)
    xf = Plan2.Range("V" & linha)

    CORTANTE = 0

    If Tipo = "Conc." Then
      If tipoApoio = "Direto" And x >= L -
xRedConc And Plan2.Range("CP19") = "SIM"
Then
        If x - xi <= 0 Then
          CORTANTE = (((L - x) / xRedConc) *
valor * (L - xi)) / L
        Else
          CORTANTE = -(((L - x) / xRedConc)
* valor * xi) / L
        End If
      End If
      If tipoApoio = "Indireto" Or x < L -
xRedConc Or Plan2.Range("CP19") = "NAO"
Then
        If x - xi <= 0 Then
          CORTANTE = (valor * (L - xi)) / L
        Else
          CORTANTE = (-valor * xi) / L
        End If
      End If
    End If

    If Tipo = "Dist." Then
      If tipoApoio = "Direto" And x >= L -
xRedDist And Plan2.Range("CP19") = "SIM" Then
        If x - xi < 0 Then CORTANTE = valor *
((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))
        If (x - xi >= 0) And (x - xf <= 0) Then
CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * ((L - xRedDist) - xi))
        If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
        End If
      End If
      If tipoApoio = "Indireto" Or x < L -
xRedDist Or Plan2.Range("CP19") = "NAO" Then
        If x - xi < 0 Then CORTANTE = valor *
((xf - xi) - ((xf ^ 2) - (xi ^ 2)) / (2 * L))

```



```

    If (x - xi >= 0) And (x - xf <= 0) Then
    CORTANTE = (valor * ((xf - xi) - ((xf ^ 2) - (xi ^
2)) / (2 * L))) - (valor * (x - xi))
    If x - xf > 0 Then CORTANTE = -valor *
(((xf ^ 2) - (xi ^ 2)) / (2 * L))
    End If
    End If
    CORTANTES(linha - 1) = CORTANTE

    Next linha
End If
'PROTENSÃO
qALIV = Plan7.Range("B6")
CortantePROT = -(qALIV * (L / 2)) + (qALIV *
x)
CORTANTES(ult) = CortantePROT

Vg_Vq = 0
For i = 0 To ult - 1
Vg_Vq = Vg_Vq + CORTANTES(i)
Next i

Vpinfinito = CORTANTES(ult)

Vsd23 = 1.4 * Vg_Vq + 0.9 * Vpinfinito

Plan2.Range("CV11") = Abs(Vsd23) 'cortante
em xf2=xi3
End Function

```

Subs

```

Public Fecha As Integer 'Autoriza o fechamento

Public Test As Integer 'testa se botao fechar foi
acionado e SalvarProjeto deve fechar o arquivo ao
terminar

Public senha As String

Public SecaoConhecida As Integer

Public SecaoCritica(2) '[nº da seção, distancia do
apoio esquerdo, tensão]

Public Ap_ef

Public Ap_est

Public Pi_ef

Public Sigma_pi

Public ep_lim_inf() 'matriz excentricidades limites
inferiores [0 = x,1 = ep], primeira linha: x=L/2

Public ep_lim_sup() 'matriz excentricidades limites
superiores [0 = x,1 = ep], primeira linha: x=L/2

```

```

Public esc 'escala do desenho da viga

Public y_DesViga 'altura escalada da viga

Public FUSO As Integer 'Testa se o fuso deve ser
desenhado

Public EQ As Integer 'Testa se o cabo equivalente
deve ser desenhado

Public RESETACABO As Integer 'Testa se os
cabos devem ser resetados (caso tenham ocorrido
mudanças no data3)

'DESENHAR LINHA (API)
Const PS_SOLID = 0
Declare PtrSafe Function GetDC& Lib "user32"
(ByVal hWnd&)
Declare PtrSafe Function LineTo& Lib "gdi32"
(ByVal hdc& _
, ByVal x&, ByVal y&)
Declare PtrSafe Function MoveToEx& Lib "gdi32"
_
(ByVal hdc&, ByVal x&, ByVal y&, lpPoint As
POINTAPI)
Declare PtrSafe Function GetActiveWindow& Lib
"user32" ()
Declare PtrSafe Function CreatePen Lib "gdi32"
(ByVal nPenStyle As Long, ByVal nWidth As
Long, ByVal crColor As Long) As Long
Declare PtrSafe Function SelectObject Lib "gdi32"
(ByVal hdc As Long, ByVal hObject As Long) As
Long
Declare PtrSafe Function DeleteObject Lib "gdi32"
(ByVal hObject As Long) As Long

Type POINTAPI
x As Long
y As Long
End Type

Dim hPen As Long
Dim hPen2 As Long
Public hdc As Long

Sub DrawLine(ByVal X1&, ByVal Y1&, ByVal
X2&, ByVal Y2&)
Dim pt As POINTAPI
MoveToEx hdc, X1, Y1, pt
LineTo hdc, X2, Y2
End Sub

Sub SalvarProjeto()
If UCase(ThisWorkbook.Name) <>
"DIMPRO.XLSM" Then
ThisWorkbook.Save
If Test = 2 Then
If Application.Workbooks.Count > 1 Then
Application.DisplayAlerts = False
Fecha = 1

```

```

        ThisWorkbook.Close
        Application.DisplayAlerts = True
    Else
        Application.DisplayAlerts = False
        Fecha = 1
        Application.Quit
    End If
End If
Else
Escolhe:
    nome = InputBox("Digite um nome para o
Projeto", "Salvar") & ".xlsm"
    If carac_especial(nome) Then
        MsgBox "Não insira caracteres especiais: /\
"" * ? < > |", vbCritical, "Atenção!"
        GoTo Escolhe
    End If
    If nome = ".xlsm" Then
        Select Case Test
            Case 0
                Exit Sub
            Case 1
                NovoProjeto
                Exit Sub
            Case 2
                Fechar
                Exit Sub
        End Select
    End If
    If UCase(nome) =
UCase(ThisWorkbook.Name) Then
        MsgBox "Escolha outro nome.",
vbOKOnly, "Nome Inválido"
        GoTo Escolhe
    Else
        Caminho = ThisWorkbook.Path & "\" &
nome
        If ExisteArquivo(nome) Then
            s = MsgBox("Já existe um arquivo com
esse nome, deseja substituí-lo?", vbYesNoCancel,
"Arquivo já existente")
            If s = vbCancel Then Exit Sub
            If s = vbYes Then
                If Aberto(nome) Then
                    MsgBox "O arquivo está aberto,
feche-o e repita esta operação.", vbOKOnly,
"Arquivo em uso"
                    GoTo Escolhe
                Else
                    Application.DisplayAlerts = False
                    ThisWorkbook.SaveAs
Filename:=Caminho
                    Application.DisplayAlerts = True
                    If Test = 2 Then
                        If Application.Workbooks.Count
> 1 Then
                            Application.DisplayAlerts =
False
                            Fecha = 1
                            ThisWorkbook.Close
                            Application.DisplayAlerts =
True
                        Else
                            Application.DisplayAlerts = False
                            Fecha = 1
                            Application.Quit
                        End If
                    Case vbYes
                        Test = 2
                        SalvarProjeto
                        Test = 0
                    End Select
                Else
                    Application.DisplayAlerts =
True
                Else
                    Application.DisplayAlerts =
False
                    Fecha = 1
                    Application.Quit
                End If
            End If
        Else
            GoTo Escolhe
        End If
    End If
    ThisWorkbook.SaveAs Filename:=Caminho
    If Test = 2 Then
        If Application.Workbooks.Count > 1
Then
            Application.DisplayAlerts = False
            Fecha = 1
            ThisWorkbook.Close
            Application.DisplayAlerts = True
        Else
            Application.DisplayAlerts = False
            Fecha = 1
            Application.Quit
        End If
    Case vbYes
        Test = 2
        SalvarProjeto
        Test = 0
    End Select
Else

```

```

If Application.Workbooks.Count > 1 Then
    Fecha = 1
    ThisWorkbook.Close
Else
    Fecha = 1
    Application.Quit
End If
End If
End Sub

Sub NovoProjeto()
    If Not ThisWorkbook.Saved Then
        r = MsgBox("Deseja salvar o Projeto atual?",
vbYesNoCancel, "Fechar Programa")
        Select Case r
            Case vbCancel
                Exit Sub
            Case vbNo

                Case vbYes
                    Test = 1
                    SalvarProjeto
                    Test = 0
                End Select
            End If
            If UCase(ThisWorkbook.Name) =
UCase("DimPro.xlsm") Then
                Plan2.Range("M17") = 0
                Limpa
                Exit Sub
            End If
            cam = ThisWorkbook.Path & "\DimPro.xlsm"
            If ExisteArquivo("DIMPRO.xlsm") Then
                If Not Aberto("dimpro.xlsm") Then
                    Application.Workbooks.Open (cam)
                    Fecha = 1
                    ThisWorkbook.Close
                    Fecha = Null
                Else
                    MsgBox "DimPro já está aberto",
vbOKOnly, "Aviso"
                    Exit Sub
                End If
            Else
                If Not Aberto("dimpro.xlsm") Then
                    Plan2.Range("M17") = 0
                    Limpa
                    ThisWorkbook.SaveAs Filename:=cam
                Else
                    MsgBox "DimPro já está aberto, feche-o e
repeita a operação", vbOKOnly, "Aviso"
                End If
            End If
        End Sub

Sub Limpa()
    Plan1.Range("A1", "K" & (Ultima_A(1, 11) +
1)).Clear
    Plan2.Range("O17").Clear

    Plan2.Range("A2", "I" & (Ultima_B(1, 9) +
1)).Clear
    Plan2.Range("R2", "AB" & (Ultima_B(18, 28) +
1)).Clear
    Plan2.Range("AE6", "AM" & (Ultima_B(31, 39)
+ 1)).Clear
    Plan2.Range("AP19", "AX" & (Ultima_B(42,
50) + 1)).Clear
    Plan2.Range("BA3", "BI" & (Ultima_B(53, 61)
+ 1)).Clear
    Plan2.Range("BJ11", "BK" & (Ultima_B(62, 63)
+ 1)).Clear
    Plan2.Range("BK9").Clear
    Plan2.Range("BK2", "BK7").Clear
    Plan2.Range("BL1", "BL" & Ultima_B(64,
64)).Clear
    Plan2.Range("BM2", "BQ" & (Ultima_B(65, 68)
+ 1)).Clear
    Plan2.Range("BS2", "BS7").Clear
    Plan2.Range("CG3", "CM" & (Ultima_B(85, 91)
+ 1)).Clear
    Plan2.Range("CA2", "CA10").Clear
    Plan3.Range("B6").Clear
    Plan3.Range("B7").Clear
    Plan3.Range("D8").Clear
    Plan3.Range("D9").Clear
    Plan3.Range("E10").Clear
    Plan3.Range("E11").Clear
    Plan3.Range("B12").Clear
    Plan3.Range("C13").Clear
    Plan3.Range("B16").Clear
    Plan3.Range("B17").Clear
    Plan3.Range("E18").Clear
    Plan3.Range("E19").Clear
    Plan3.Range("E20").Clear
    Plan3.Range("C23").Clear
    Plan3.Range("C24", "E24").Clear
    Plan3.Range("C27").Clear
    For i = 28 To 37
        If i = 35 Then
            Plan3.Range("C35").Clear
        Else
            Plan3.Range("B" & i).Clear
        End If
    Next i
    Plan3.Range("C38").Clear
    Plan3.Range("D41").Clear
    Plan3.Range("D42").Clear
    Plan3.Range("B44").Clear
    Plan3.Range("B48").Clear
    Plan3.Range("B49").Clear
    Plan3.Range("A51").Clear

    Dim wkb As Workbook

    Application.DisplayAlerts = False
    Set wkb = ThisWorkbook

    If Len(Trim("ELS")) = 0 Then GoTo
PROC_EXIT

```

```

    On Error Resume Next
    wkb.Worksheets("ELS").Delete
    wkb.Worksheets("ELU").Delete
    Set wkb = Nothing

    Application.DisplayAlerts = True

PROC_EXIT:
End Sub

Sub Imprimir()
    ActiveSheet.PageSetup.PrintArea = "$A$1:$K$"
    & Ultima(11)
    ActiveSheet.PrintPreview
End Sub

Sub Relatorio()
    Dim wdApp As Object, wdDoc As Object
    Set wdApp = CreateObject("Word.Application")
    Set wdDoc = wdApp.Documents.Add
    wdApp.Visible = True
    wdApp.Activate
    ActiveSheet.Range("A1", "K" &
    Ultima(1)).Copy
    wdApp.activedocument.PageSetup.LeftMargin =
    20
    wdApp.activedocument.Range(0).Paste
    If UCase(ThisWorkbook.Name) =
    UCase("DimPro.xlsm") Then
        For i = 1 To 100
            Arq = "Relatorio" & i & ".doc"
            If Not ExisteArquivo(Arq) Then Exit For
        Next i
        cam = ThisWorkbook.Path & "\" & Arq
        wdApp.activedocument.SaveAs cam
    Else
        nome = Split(ThisWorkbook.Name, ".")
        Arq = nome(0) & ".doc"
        If ExisteArquivo(Arq) Then
            For i = 1 To 100
                Arq = nome(0) & i & ".doc"
                If Not ExisteArquivo(Arq) Then Exit For
            Next i
            cam = ThisWorkbook.Path & "\" & Arq
            wdApp.activedocument.SaveAs cam
        Else
            cam = ThisWorkbook.Path & "\" & Arq
            wdApp.activedocument.SaveAs cam
        End If
    End If
    Set wdApp = Nothing
    Set wdDoc = Nothing
End Sub

Sub GerarPlanilhaELS(strNomePlan As String)
    'Usada para gerar planilhas ELS

    Dim wkb As Workbook
    Dim wks As Worksheet

    Application.DisplayAlerts = False
    Set wkb = ThisWorkbook

    If Len(Trim(strNomePlan)) = 0 Then GoTo
    PROC_EXIT

    On Error Resume Next
    wkb.Worksheets(strNomePlan).Delete
    wkb.Worksheets.Add
    After:=wkb.Worksheets(3), Count:=1
    Set wks = wkb.Worksheets(4)
    wks.Name = strNomePlan

    Set wks = Nothing
    Set wkb = Nothing

    Application.DisplayAlerts = True

PROC_EXIT:
End Sub

Sub GerarPlanilhaELU(strNomePlan As String)
    'Usada para gerar planilhas ELU

    Dim wkb As Workbook
    Dim wks As Worksheet

    Application.DisplayAlerts = False
    Set wkb = ThisWorkbook

    If Len(Trim(strNomePlan)) = 0 Then GoTo
    PROC_EXIT

    On Error Resume Next
    wkb.Worksheets(strNomePlan).Delete
    wkb.Worksheets.Add
    After:=wkb.Worksheets(4), Count:=1
    Set wks = wkb.Worksheets(5)
    wks.Name = strNomePlan

    Set wks = Nothing
    Set wkb = Nothing

    Application.DisplayAlerts = True

PROC_EXIT:
End Sub

Sub GerarPlanFLECHA(strNomePlan As String)

    Dim wkb As Workbook
    Dim wks As Worksheet

    Application.DisplayAlerts = False
    Set wkb = ThisWorkbook

    If Len(Trim(strNomePlan)) = 0 Then GoTo
    PROC_EXIT

    On Error Resume Next
    wkb.Worksheets(strNomePlan).Delete

```

```

    wkb.Worksheets.Add
After:=wkb.Worksheets(wkb.Worksheets.Count)
    Set wks =
wkb.Worksheets(wkb.Worksheets.Count)
    wks.Name = strNomePlan

    Set wks = Nothing
    Set wkb = Nothing

    Application.DisplayAlerts = True

PROC_EXIT:
End Sub

Sub Atualiza_LB_Secao()
    ult = Ultima_B(1, 1)
    If ult = 1 Then
        Data1.ListBox1.RowSource =
"BIBLIOTECA!A2:E2"
    Else
        Data1.ListBox1.RowSource =
"BIBLIOTECA!A2:E" & ult
    End If
End Sub

Sub Atualiza_LB_Carga()
    ult = Ultima_B(18, 18)
    If ult = 1 Then
        Data2.ListBox1.RowSource =
"BIBLIOTECA!R2:Z2"
    Else
        Data2.ListBox1.RowSource =
"BIBLIOTECA!R2:Z" & ult
    End If
End Sub

Sub Atualiza_LB_Cabos()
    ult = Ultima_B(62, 62)
    If ult = 10 Then
        Data3.ListBox2.RowSource =
"BIBLIOTECA!BJ11:BK11"
    Else
        Data3.ListBox2.RowSource =
"BIBLIOTECA!BJ11:BK" & ult
    End If
End Sub

Sub Secao_Critica()

    Tensao_Critica = 0
    N = Plan3.Range("C13")
    L = Plan3.Range("B12")
    NivelProt = Plan3.Range("C24")

    If NivelProt = "Nível 3 - Protensão Completa"
Then
        For i = 1 To N
            x = ((i - 1) * L) / (N - 1)
            MatrizTensoes (x)
            CR = Comb_Rara_Inferior()
            If CR > Tensao_Critica Then
                Tensao_Critica = CR
                dist = x
                secao = i
            End If
        Next i
        x = L / 2
        MatrizTensoes (x)
        CR = Comb_Frequente_Inferior()
        If CR > Tensao_Critica Then
            Tensao_Critica = CR
            dist = x
            secao = "Meio do Vão"
        End If
    End If
    SecaoCritica(0) = secao
    SecaoCritica(1) = dist

    If NivelProt = "Nível 2 - Protensão Limitada"
Then
        For i = 1 To N
            x = ((i - 1) * L) / (N - 1)
            MatrizTensoes (x)
            CF = Comb_Frequente_Inferior()
            If CF > Tensao_Critica Then
                Tensao_Critica = CF
                dist = x
                secao = i
            End If
        Next i
        x = L / 2
        MatrizTensoes (x)
        CF = Comb_Frequente_Inferior()
        If CF > Tensao_Critica Then
            Tensao_Critica = CF
            dist = x
            secao = "Meio do Vão"
        End If
    End If
    If NivelProt = "Nível 1 - Protensão Parcial"
Then
        For i = 1 To N
            x = ((i - 1) * L) / (N - 1)
            MatrizTensoes (x)
            CF = Comb_Frequente_Inferior()
            If CF > Tensao_Critica Then
                Tensao_Critica = CF
                dist = x
                secao = i
            End If
        Next i
        x = L / 2
        MatrizTensoes (x)
        CF = Comb_Frequente_Inferior()
        If CF > Tensao_Critica Then
            Tensao_Critica = CF
            dist = x
            secao = "Meio do Vão"
        End If
    End If

```

```

SecaoCritica(2) = Tensao_Critica
Plan3.Range("C27") = dist

End Sub

Sub Estima_P()

NivelProt = Plan3.Range("C24")
a = Plan3.Range("B7")
Wi = Plan3.Range("E10")
Ws = Plan3.Range("E11")
e = Plan3.Range("B28")

Secao_Critica
x = SecaoCritica(1)
MatrizTensoes (x)

If NivelProt = "Nível 3 - Protensão Completa"
Then
    CF = Comb_Frequente_Inferior()
    CR = Comb_Rara_Inferior()
    CRS = Comb_Rara_Superior()

    'ELS-D
    N1 = (-CF) / ((1 / a) + (e / Wi))

    'ELS-F
    fck = Plan3.Range("B16")
    fator = Plan2.Range("O17")
    fctf = fator * 0.7 * 0.3 * (fck ^ (2 / 3)) * 1000

    N2 = (fctf - CR) / ((1 / a) + (e / Wi))

    If N1 < N2 Then
        N = N1
    Else
        N = N2
    End If

    'ELS-CE
    TensaoSup = CRS + (N / a) + ((N * e) / Ws)
    If TensaoSup < (-0.6 * fck * 1000) Then
        MsgBox "Estado Limite de Compressão
Excessiva atingido!", vbOKOnly, "Atenção!"
        Plan3.Range("D28") = "Atenção!"
        Plan3.Range("D29") = "Estado Limite de
Compressão Excessiva atingido!"
        Plan3.Range("D28:K29").Interior.Color =
RGB(255, 255, 0)
    End If
    End If

    If NivelProt = "Nível 1 - Protensão Parcial" Then
        CRS = Comb_Rara_Superior()
        'Pinf_est = (momentopermbalanc/ep)
        n = Plan6.Range("B7") / Plan3.Range("B28")

        'ELS-CE
        'TensaoSup = CRS + (N / a) + ((N * e) / Ws)
        'If TensaoSup < (-0.6 * fck * 1000) Then
        'MsgBox "Estado Limite de Compressão
Excessiva atingido!", vbOKOnly, "Atenção!"
        'Plan3.Range("D28") = "Atenção!"
        'Plan3.Range("D29") = "Estado Limite de
Compressão Excessiva atingido!"
        'Plan3.Range("D28:D29").Interior.Color =
RGB(255, 255, 0)
        'End If
        End If

        Pinf_est = n
        Perda_Est = Plan3.Range("B30")
        Pi_est = Pinf_est / ((100 - Perda_Est) / 100)

        Plan3.Range("B29") = Abs(Pinf_est)
        Plan3.Range("B31") = Abs(Pi_est)
    End Sub

    Sub FusoLimite_Matriz_ep()

        Nsec = Plan3.Range("C13")
        ReDim ep_lim_inf(0 To Nsec, 0 To 1) 'primeira
linha é para x=L/2
        ReDim ep_lim_sup(0 To Nsec, 0 To 1)

        L = Plan3.Range("B12")

```

```

h = Plan3.Range("B6")
Ycg = Plan3.Range("D8")
ult = Ultima_B(18, 18)
fck = Plan3.Range("B16")
fator = Plan2.Range("O17")
Pi = Plan3.Range("B37")
PerdaEst = Plan3.Range("B30")
Pinf = Plan3.Range("B37") * (100 - PerdaEst) /
100
a = Plan3.Range("B7")
Wi = Plan3.Range("E10")
Ws = Plan3.Range("E11")
fctf = fator * 0.7 * 0.3 * (fck ^ (2 / 3)) * 1000
fckj = Plan3.Range("B17")
NivelProt = Plan3.Range("C24")

If Pi = 0 Then Exit Sub

'ELU-PROTENSÃO EM VAZIO -
EXCENTRICIDADE LIMITE DO CENTRO DE
PRESSÃO
If Plan3.Range("C23") = "Pré-Tração" Then
    N = -Pi
Else
    N = -Pi * 1.1
End If
eki = -(Wi / a)
eks = -(Ws / a)
aiv = eki * (1 - ((-0.7 * fckj * 1000) / (N / a)))
asv = eks * (1 - ((1.2 * 0.3 * (fckj ^ (2 / 3)) *
1000) / (N / a)))
If aiv < asv Then
    av = aiv
Else
    av = asv
End If

N = -Pinf
For i = 1 To Nsec 'calcula as matrizes de ep's
para cada seção

    x = ((i - 1) * L) / (Nsec - 1)
    MatrizTensoes (x)

'ELU-PROTENSÃO EM VAZIO - LIMITE
INFERIOR
emg1 = (TENSAO(0, 0) * Wi) / (Pi * 1.1)
ei = av + emg1
If ei > Ycg Then ei = Ycg
If ei < (-h - Ycg) Then ei = (-h - Ycg)
ep_lim_inf(i, 0) = x
ep_lim_inf(i, 1) = ei

'ELS - LIMITE SUPERIOR
If NivelProt = "Nível 3 - Protensão Completa"
Then
    CF = Comb_Frequente_Inferior()
    CR = Comb_Rara_Inferior()
    CRS = Comb_Rara_Superior()

'ELS-D
ais_CF = eki
emgq1 = (CF * Wi) / Pinf
e1 = ais_CF + emgq1

'ELS-F
ais_CR = eki * (1 - (fctf / (N / a)))
emgq2 = (CR * Wi) / Pinf
e2 = ais_CR + emgq2

'ELS-CE
Ass = eks * (1 - ((-0.6 * fck * 1000) / (N /
a)))
emgq3 = (CRS * Ws) / Pinf
e3 = Ass + emgq3

If e1 >= e2 And e1 >= e3 Then Es = e1
If e2 >= e1 And e2 >= e3 Then Es = e2
If e3 >= e1 And e3 >= e2 Then Es = e3
If Es > Ycg Then Es = Ycg
If Es < (-h - Ycg) Then Es = (-h - Ycg)
ep_lim_sup(i, 0) = x
ep_lim_sup(i, 1) = Es
End If

If NivelProt = "Nível 2 - Protensão Limitada"
Then
    CF = Comb_Frequente_Inferior()
    CQP = Comb_QuasePermanente_Inferior()
    CFS = Comb_Frequente_Superior()

'ELS-D
ais_CQP = eki
emgq1 = (CQP * Wi) / Pinf
e1 = ais_CQP + emgq1

'ELS-F
ais_CF = eki * (1 - (fctf / (N / a)))
emgq2 = (CF * Wi) / Pinf
e2 = ais_CF + emgq2

'ELS-CE
Ass = eks * (1 - ((-0.6 * fck * 1000) / (N /
a)))
emgq3 = (CFS * Ws) / Pinf
e3 = Ass + emgq3

If e1 >= e2 And e1 >= e3 Then Es = e1
If e2 >= e1 And e2 >= e3 Then Es = e2
If e3 >= e1 And e3 >= e2 Then Es = e3
If Es > Ycg Then Es = Ycg
If Es < (-h - Ycg) Then Es = (-h - Ycg)
ep_lim_sup(i, 0) = x
ep_lim_sup(i, 1) = Es
End If

If NivelProt = "Nível 1 - Protensão Parcial"
Then
    CF = Comb_Frequente_Inferior()
    CQP = Comb_QuasePermanente_Inferior()

```

```

CFS = Comb_Frequente_Superior()

'ELS-D
ais_CQP = eki
emgq1 = (CQP * Wi) / Pinf
e1 = ais_CQP + emgq1

'ELS-F
ais_CF = eki * (1 - (fctf / (N / a)))
emgq2 = (CF * Wi) / Pinf
e2 = ais_CF + emgq2

'ELS-CE
Ass = eks * (1 - ((-0.6 * fck * 1000) / (N /
a)))
emgq3 = (CFS * Ws) / Pinf
e3 = Ass + emgq3

If e1 >= e2 And e1 >= e3 Then Es = e1
If e2 >= e1 And e2 >= e3 Then Es = e2
If e3 >= e1 And e3 >= e2 Then Es = e3
If Es > Ycg Then Es = Ycg
If Es < (-(h - Ycg)) Then Es = (-(h - Ycg))
ep_lim_sup(0, 0) = x
ep_lim_sup(0, 1) = Es
End If

If NivelProt = "Nível 2 - Protensão Limitada"
Then
CF = Comb_Frequente_Inferior()
CQP = Comb_QuasePermanente_Inferior()
CFS = Comb_Frequente_Superior()
'ELS-D
ais_CQP = eki
emgq1 = (CQP * Wi) / Pinf
e1 = ais_CQP + emgq1

'ELS-F
ais_CF = eki * (1 - (fctf / (N / a)))
emgq2 = (CF * Wi) / Pinf
e2 = ais_CF + emgq2

'ELS-CE
Ass = eks * (1 - ((-0.6 * fck * 1000) / (N / a)))
emgq3 = (CFS * Ws) / Pinf
e3 = Ass + emgq3

If e1 >= e2 And e1 >= e3 Then Es = e1
If e2 >= e1 And e2 >= e3 Then Es = e2
If e3 >= e1 And e3 >= e2 Then Es = e3
If Es > Ycg Then Es = Ycg
If Es < (-(h - Ycg)) Then Es = (-(h - Ycg))
ep_lim_sup(0, 0) = x
ep_lim_sup(0, 1) = Es
End If

If NivelProt = "Nível 1 - Protensão Parcial" Then
CF = Comb_Frequente_Inferior()
CQP = Comb_QuasePermanente_Inferior()
CFS = Comb_Frequente_Superior()

'ELS-D
ais_CQP = eki
emgq1 = (CQP * Wi) / Pinf
e1 = ais_CQP + emgq1

'ELS-F
ais_CF = eki * (1 - (fctf / (N / a)))
emgq2 = (CF * Wi) / Pinf
e2 = ais_CF + emgq2

'ELS-CE
Ass = eks * (1 - ((-0.6 * fck * 1000) / (N / a)))
emgq3 = (CFS * Ws) / Pinf

```



```

e3 = Ass + emgq3

If e1 >= e2 And e1 >= e3 Then Es = e1
If e2 >= e1 And e2 >= e3 Then Es = e2
If e3 >= e1 And e3 >= e2 Then Es = e3
If Es > Ycg Then Es = Ycg
If Es < (-(h - Ycg)) Then Es = (-(h - Ycg))
ep_lim_sup(0, 0) = x
ep_lim_sup(0, 1) = Es
End If
End Sub

Sub Estima_Cabos()
L = Plan3.Range("B12")
ult = Ultima_B(65, 69)
If ult > 1 Then Plan2.Range("BM2", "BQ" &
ult).Clear
ultC = Ultima_B(62, 62)
num_cabos = Plan2.Range("BJ" & ultC)

ei = ep_lim_sup(1, 1) + ((ep_lim_inf(1, 1) -
ep_lim_sup(1, 1)) / 2)
xm = L / 2
em = ep_lim_sup(0, 1) + ((ep_lim_inf(0, 1) -
ep_lim_sup(0, 1)) / 2)
If num_cabos > 0 And num_cabos <> "" Then
For i = 1 To num_cabos
Plan2.Range("BM" & (i + 1)) = i
Plan2.Range("BN" & (i + 1)) =
Application.WorksheetFunction.RoundUp(ei, 4)
Plan2.Range("BO" & (i + 1)) =
Application.WorksheetFunction.RoundUp(xm, 4)
Plan2.Range("BP" & (i + 1)) =
Application.WorksheetFunction.RoundUp(em, 4)
Plan2.Range("BQ" & (i + 1)) =
Plan2.Range("BK" & (i + 10))
Next i
End If
End Sub

Sub DesenhaViga(x0 As Integer, y0 As Integer, xf
As Integer, escala, ymax As Integer)
L = Plan3.Range("B12")
h = Plan3.Range("B6")
Ycg = Plan3.Range("D8")
y = (h / L * 700) * escala
Do While y > ymax
escala = escala - 0.1
y = (h / L * 700) * escala
Loop
esc = escala
If Len(esc) >= 5 Then esc = Format(escala,
"0.00")
y_DesViga = y
hdc = GetDC(GetActiveWindow)
DoEvents
'retangulo
DrawLine x0, y0, xf, y0 'linha inferior
DrawLine x0, y0, x0, y0 - y 'linha da esquerda
DrawLine x0, y0 - y, xf, y0 - y 'linha superior
DrawLine xf, y0, xf, y0 - y 'linha da direita
'triangulo esquerda
DrawLine x0 + 10, y0, x0, y0 + 20
DrawLine x0, y0 + 20, x0 + 20, y0 + 20
DrawLine x0 + 10, y0, x0 + 20, y0 + 20
'triangulo direita
DrawLine xf - 10, y0, xf, y0 + 20
DrawLine xf, y0 + 20, xf - 20, y0 + 20
DrawLine xf - 10, y0, xf - 20, y0 + 20
'centro de gravidade
hPen = CreatePen(1, 1, RGB(0, 0, 0))
DeleteObject SelectObject(hdc, hPen)
DrawLine x0, y0 - (y * Ycg / h), xf, y0 - (y *
Ycg / h)

DeleteObject hPen
End Sub
Sub DesenhaFuso(x0 As Integer, y0 As Integer, xf
As Integer, escala)
L = Plan3.Range("B12")
h = Plan3.Range("B6")
Ycg = Plan3.Range("D8")
Nsec = Plan3.Range("C13")
hdc = GetDC(GetActiveWindow)
DoEvents
Data3.Label20.ForeColor = RGB(255, 0, 0)
Data3.Label20.Visible = False
Plan3.Range("E32:K33").Interior.Color =
RGB(255, 255, 255)
Plan3.Range("E32:K33").Clear
If Plan3.Range("C35") < 0 Or
Plan3.Range("C35") = "" Then Exit Sub
FusoLimite_Matriz_ep
For i = 1 To Nsec - 1
hPen = CreatePen(PS_SOLID, 2, RGB(0, 0,
0))
DeleteObject SelectObject(hdc, hPen)

Xanterior_esc = ((ep_lim_inf(i, 0) / L) * 700)
+ x0
Xatual_esc = ((ep_lim_inf(i + 1, 0) / L) * 700)
+ x0
Yanterior_inf_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_inf(i, 1) / h) * y_DesViga))
Yanterior_sup_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_sup(i, 1) / h) * y_DesViga))
Yatual_inf_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_inf(i + 1, 1) / h) *
y_DesViga))
Yatual_sup_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_sup(i + 1, 1) / h) *
y_DesViga))

'LIMITE INFERIOR
DrawLine Xanterior_esc, Yanterior_inf_esc,
Xatual_esc, Yatual_inf_esc
'LIMITE SUPERIOR
DrawLine Xanterior_esc,
Yanterior_sup_esc, Xatual_esc, Yatual_sup_esc

```

```

    If ep_lim_inf(i, 1) >= ep_lim_sup(i, 1) Then
        hPen3 = CreatePen(PS_SOLID, 3, RGB(0,
200, 0))
        DeleteObject SelectObject(hdc, hPen3)
        DrawLine ((ep_lim_inf(i, 0) / L) * 700) +
x0, y0 - (((Ycg / h) * y_DesViga) - ((ep_lim_inf(i,
1) / h) * y_DesViga)), ((ep_lim_inf(i, 0) / L) * 700)
+ x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_sup(i, 1) / h) * y_DesViga))
    Else
        hPen2 = CreatePen(PS_SOLID, 3, vbRed)
        DeleteObject SelectObject(hdc, hPen2)
        DrawLine ((ep_lim_inf(i, 0) / L) * 700) +
x0, y0 - (((Ycg / h) * y_DesViga) - ((ep_lim_inf(i,
1) / h) * y_DesViga)), ((ep_lim_inf(i, 0) / L) * 700)
+ x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_sup(i, 1) / h) * y_DesViga))
        Data3.Label20.Visible = True
        Plan3.Range("E32") = "Atenção!"
        Plan3.Range("E33") = "Traçado do cabo
equivalente impossível!"
        Plan3.Range("E32:K33").Interior.Color =
RGB(255, 0, 0)
    End If
    Next i
    'ultima seção
    If ep_lim_inf(Nsec, 1) >= ep_lim_sup(Nsec, 1)
Then
        hPen3 = CreatePen(PS_SOLID, 3, RGB(0,
200, 0))
        DeleteObject SelectObject(hdc, hPen3)
        DrawLine ((ep_lim_inf(Nsec, 0) / L) * 700) +
x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_inf(Nsec, 1) / h) * y_DesViga)),
((ep_lim_inf(Nsec, 0) / L) * 700) + x0, y0 - (((Ycg /
h) * y_DesViga) - ((ep_lim_sup(Nsec, 1) / h) *
y_DesViga))
    Else
        hPen2 = CreatePen(PS_SOLID, 3, vbRed)
        DeleteObject SelectObject(hdc, hPen2)
        DrawLine ((ep_lim_inf(Nsec, 0) / L) * 700) +
x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_inf(Nsec, 1) / h) * y_DesViga)),
((ep_lim_inf(Nsec, 0) / L) * 700) + x0, y0 - (((Ycg /
h) * y_DesViga) - ((ep_lim_sup(Nsec, 1) / h) *
y_DesViga))
        Data3.Label20.Visible = True
        Plan3.Range("E32") = "Atenção!"
        Plan3.Range("E33") = "Traçado do cabo
equivalente impossível!"
        Plan3.Range("E32:K33").Interior.Color =
RGB(255, 0, 0)
    End If
    'seção no meio do vão
    If Application.WorksheetFunction.IsEven(Nsec)
Then
        If ep_lim_inf(0, 1) >= ep_lim_sup(0, 1) Then
            hPen3 = CreatePen(PS_SOLID, 3, RGB(0,
200, 0))
            DeleteObject SelectObject(hdc, hPen3)

```

```

        DrawLine ((ep_lim_inf(0, 0) / L) * 700) +
x0, y0 - (((Ycg / h) * y_DesViga) - ((ep_lim_inf(0,
1) / h) * y_DesViga)), ((ep_lim_inf(0, 0) / L) * 700)
+ x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_sup(0, 1) / h) * y_DesViga))
    Else
        hPen2 = CreatePen(PS_SOLID, 3, vbRed)
        DeleteObject SelectObject(hdc, hPen2)
        DrawLine ((ep_lim_inf(0, 0) / L) * 700) +
x0, y0 - (((Ycg / h) * y_DesViga) - ((ep_lim_inf(0,
1) / h) * y_DesViga)), ((ep_lim_inf(0, 0) / L) * 700)
+ x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_sup(0, 1) / h) * y_DesViga))
        Data3.Label20.Visible = True
        Plan3.Range("E32") = "Atenção!"
        Plan3.Range("E33") = "Traçado do cabo
equivalente impossível!"
        Plan3.Range("E32:K33").Interior.Color =
RGB(255, 0, 0)
    End If
    End If
    DeleteObject hPen
    DeleteObject hPen2
    DeleteObject hPen3
    hdc = GetDC(GetActiveWindow)
End Sub

Sub DesenhaCabos(x0 As Integer, y0 As Integer, xf
As Integer, escala)
    If Plan3.Range("C35") < 0 Or
Plan3.Range("C35") = "" Then Exit Sub
    ultC = Ultima_B(62, 62)
    num_cabos = Plan2.Range("BJ" & ultC)
    L = Plan3.Range("B12")
    h = Plan3.Range("B6")
    Ycg = Plan3.Range("D8")
    Nsec = Plan3.Range("C13")
    hdc = GetDC(GetActiveWindow)
    DoEvents
    Tracado_Cabos
    For Cabo = 1 To num_cabos
        For i = 1 To Nsec - 1
            hPen = CreatePen(PS_SOLID, 1, RGB(168,
168, 168))
            DeleteObject SelectObject(hdc, hPen)
            Xanterior_esc = (((i - 1) * L) / (Nsec - 1)) /
L) * 700) + x0
            Xatual_esc = (((i + 1) - 1) * L) / (Nsec -
1)) / L) * 700) + x0
            Yanterior_esc = y0 - (((Ycg / h) *
y_DesViga) - ((CABOS(Cabo, i) / h) *
y_DesViga))
            Yatual_esc = y0 - (((Ycg / h) * y_DesViga)
- ((CABOS(Cabo, i + 1) / h) * y_DesViga))

            DrawLine Xanterior_esc, Yanterior_esc,
Xatual_esc, Yatual_esc
        Next i
    Next Cabo
    DeleteObject hPen

```

```

End Sub

Sub DesenhaFuso_2(x0 As Integer, y0 As Integer,
xf As Integer, escala)
    L = Plan3.Range("B12")
    h = Plan3.Range("B6")
    Ycg = Plan3.Range("D8")
    Nsec = Plan3.Range("C13")
    hdc = GetDC(GetActiveWindow)
    DoEvents
    FusoLimite_Matriz_ep
    If Plan3.Range("C35") < 0 Or
Plan3.Range("C35") = "" Then Exit Sub
    For i = 1 To Nsec - 1
        hPen = CreatePen(PS_SOLID, 2, RGB(35,
142, 35))
        DeleteObject SelectObject(hdc, hPen)

        Xanterior_esc = ((ep_lim_inf(i, 0) / L) * 700)
+ x0
        Xatual_esc = ((ep_lim_inf(i + 1, 0) / L) * 700)
+ x0
        Yanterior_inf_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_inf(i, 1) / h) * y_DesViga))
        Yanterior_sup_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_sup(i, 1) / h) * y_DesViga))
        Yatual_inf_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_inf(i + 1, 1) / h) *
y_DesViga))
        Yatual_sup_esc = y0 - (((Ycg / h) *
y_DesViga) - ((ep_lim_sup(i + 1, 1) / h) *
y_DesViga))

        'LIMITE INFERIOR
        DrawLine Xanterior_esc, Yanterior_inf_esc,
Xatual_esc, Yatual_inf_esc
        'LIMITE SUPERIOR
        DrawLine Xanterior_esc, Yanterior_sup_esc,
Xatual_esc, Yatual_sup_esc
        Next i
        DrawLine ((ep_lim_inf(Nsec, 0) / L) * 700) + x0,
y0 - (((Ycg / h) * y_DesViga) - ((ep_lim_inf(Nsec,
1) / h) * y_DesViga)), ((ep_lim_inf(Nsec, 0) / L) *
700) + x0, y0 - (((Ycg / h) * y_DesViga) -
((ep_lim_sup(Nsec, 1) / h) * y_DesViga))
        DrawLine ((ep_lim_inf(1, 0) / L) * 700) + x0, y0
- (((Ycg / h) * y_DesViga) - ((ep_lim_inf(1, 1) / h)
* y_DesViga)), ((ep_lim_inf(1, 0) / L) * 700) + x0,
y0 - (((Ycg / h) * y_DesViga) - ((ep_lim_sup(1, 1) /
h) * y_DesViga))
        DeleteObject hPen
        hdc = GetDC(GetActiveWindow)
    End Sub

Sub Cabo_Equivalente()
    Nsec = Plan3.Range("C13")
    TotalFios = Plan3.Range("C35")
    ultC = Ultima_B(62, 62)
    If TotalFios = 0 Or ultC = 10 Then Exit Sub
    num_cabos = Plan2.Range("BJ" & ultC)

    L = Plan3.Range("B12")
    ReDim EQUIVALENTE(0 To Nsec, 0 To 1)
    Plan3.Range("E34:K35").Interior.Color =
RGB(255, 255, 255)
    Plan3.Range("E34:K35").Clear
    FusoLimite_Matriz_ep
    Tracado_Cabos
    fora = 0
    If Application.WorksheetFunction.IsEven(Nsec)
Then
        num_sec = Nsec / 2
    Else
        num_sec = (Nsec + 1) / 2
    End If
    For i = 0 To num_sec
        somatorio = 0
        somatorio_alfa = 0
        If i = 0 Then
            x = L / 2
        Else
            x = (((i - 1) * L) / (Nsec - 1))
        End If
        For j = 1 To num_cabos
            num_fios = Plan2.Range("BK" & (j + 10))
            somatorio = somatorio + (CABOS(j, i) *
num_fios)
            linha = j + 1
            ei = Plan2.Range("BN" & linha)
            xm = Plan2.Range("BO" & linha)
            em = Plan2.Range("BP" & linha)
            If (x - xm) < 0 And xm <> 0 Then tg_alfa =
((2 * (-em + ei) / (xm ^ 2)) * x) + (2 * (em - ei) /
xm)
            If (x - xm) >= 0 Or xm = 0 Then tg_alfa = 0
            alfa = Atn(tg_alfa) 'radianos
            somatorio_alfa = somatorio_alfa + (alfa *
num_fios)
        Next j
        EQUIVALENTE(i, 0) = somatorio / TotalFios
        EQUIVALENTE(i, 1) = somatorio_alfa /
TotalFios
        If i > 0 Then EQUIVALENTE(Nsec + 1 - i, 0)
= somatorio / TotalFios
        If i > 0 Then EQUIVALENTE(Nsec + 1 - i, 1)
= (somatorio_alfa / TotalFios) * (-1)
        If (EQUIVALENTE(i, 0) < ep_lim_sup(i, 1)
Or EQUIVALENTE(i, 0) > ep_lim_inf(i, 1)) And
fora = 0 Then
            MsgBox "Cabo equivalente fora do fuso
limite!", vbOKOnly, "Atenção!"
            Plan3.Range("E34") = "Atenção!"
            Plan3.Range("E35") = "Cabo equivalente
fora do fuso limite!"
            Plan3.Range("E34:K35").Interior.Color =
RGB(255, 0, 0)
            fora = 1
        End If
    Next i
    fora = 0
End Sub

```

```

Sub DesenhaEquivalente(x0 As Integer, y0 As Integer, xf As Integer, escala)
    ultC = Ultima_B(62, 62)
    If ultC = 10 Then Exit Sub
    num_cabos = Plan2.Range("BJ" & ultC)
    L = Plan3.Range("B12")
    h = Plan3.Range("B6")
    Ycg = Plan3.Range("D8")
    Nsec = Plan3.Range("C13")
    hdc = GetDC(GetActiveWindow)
    DoEvents
    'Cabo_Equivalente
    For i = 1 To Nsec - 1
        hPen = CreatePen(PS_SOLID, 3, RGB(56, 176, 222))
        DeleteObject SelectObject(hdc, hPen)
        Xanterior_esc = (((i - 1) * L) / (Nsec - 1)) / L * 700 + x0
        Xatual_esc = (((i + 1) - 1) * L) / (Nsec - 1) / L * 700 + x0
        Yanterior_esc = y0 - ((Ycg / h) * y_DesViga) - ((EQUIVALENTE(i, 0) / h) * y_DesViga)
        Yatual_esc = y0 - ((Ycg / h) * y_DesViga) - ((EQUIVALENTE(i + 1, 0) / h) * y_DesViga)

        DrawLine Xanterior_esc, Yanterior_esc, Xatual_esc, Yatual_esc
    Next i
    DeleteObject hPen
End Sub

Sub PERDA_ATRITO()
    Nsec = Plan3.Range("C13")
    L = Plan3.Range("B12")
    u = Plan3.Range("D41")
    b = Plan3.Range("D42")
    p = Plan3.Range("B37")
    ReDim ATRITO(0 To Nsec)
    If Plan3.Range("C23") = "Pré-Tração" Then
        For i = 0 To Nsec
            ATRITO(i) = p
        Next i
        Plan3.Range("D41") = "-"
        Plan3.Range("D42") = "-"
    Else
        Cabo_Equivalente
        If Plan3.Range("A51") = "Ativa-Ativa" Then
            If
                Application.WorksheetFunction.IsEven(Nsec) Then
                    num_sec = Nsec / 2
                Else
                    num_sec = (Nsec + 1) / 2
                End If
            For i = 0 To num_sec
                If i = 0 Then
                    x = L / 2
                Else
                    x = (((i - 1) * L) / (Nsec - 1))
                End If
                If i = 0 Then delta_alfa = EQUIVALENTE(1, 1)
                If i = 1 Then delta_alfa = 0
                If i > 1 Then delta_alfa = delta_alfa + Abs(EQUIVALENTE(i - 1, 1) - EQUIVALENTE(i, 1))
                k = -(u * (delta_alfa + (b * x)))
                P_atrito = p * Exp(k)
                ATRITO(i) = P_atrito
                If i > 0 Then ATRITO(Nsec + 1 - i) = P_atrito
            Next i
            'MsgBox "S: " & i & " x: " & x & " P: " & ATRITO(i), vbOKOnly
        End If
    End Sub

Sub PERDA_ANCORAGEM()
    Nsec = Plan3.Range("C13")
    L = Plan3.Range("B12")
    DeltaL = Plan3.Range("B44") * (10 ^ (-3))
    Ap = Plan3.Range("B36")
    Ep = Plan3.Range("E20")
    ReDim ANCORAGEM(0 To Nsec)
    If Plan3.Range("C23") = "Pré-Tração" Then
        Delta_P = (DeltaL / L) * (Ep / 10) * Ap
        For k = 0 To Nsec
            ANCORAGEM(k) = ATRITO(k) - Delta_P
        Next k
    Else
        A_delta = DeltaL * (Ep / 10) * Ap
        A_indesloc = 0
        If Plan3.Range("A51") = "Ativa-Ativa" Then
            If
                Application.WorksheetFunction.IsEven(Nsec) Then
                    num_sec = Nsec / 2
                Else
                    num_sec = (Nsec + 1) / 2 'já inclui o meio do vao
            End If
            For i = 2 To num_sec
                x = (((i - 1) * L) / (Nsec - 1))
            Next i
        End If
    End Sub

```

```

x_ant = (((i - 1) - 1) * L) / (Nsec - 1)
A_indesloc_ant = A_indesloc
A_indesloc = A_indesloc + (((ATRITO(i
- 1) - ATRITO(i)) * (x - x_ant) / 2) + ((ATRITO(i -
1) - ATRITO(i)) * x_ant)) * 2
If A_indesloc >= A_delta Then
Exit For
End If
If i = num_sec And
Application.WorksheetFunction.IsEven(Nsec) Then
x = L / 2
x_ant = (((num_sec - 1) * L) / (Nsec -
1))
A_indesloc_ant = A_indesloc
A_indesloc = A_indesloc +
(((ATRITO(num_sec) - ATRITO(0)) * (x - x_ant) /
2) + ((ATRITO(num_sec) - ATRITO(0)) * x_ant))
* 2
If A_indesloc >= A_delta Then
i = 0
Exit For
End If
End If
Next i

If A_indesloc = A_delta Then 'Ponto está
antes do (ou no) meio do vão e coincide com uma
seção i
L0 = x
For k = 0 To num_sec
If k = 0 Then
ANCORAGEM(k) = ATRITO(k)
Else
If k < i Then
ANCORAGEM(k) = ATRITO(k)
- (2 * (ATRITO(k) - ATRITO(i)))
ANCORAGEM(Nsec + 1 - k) =
ANCORAGEM(k)
Else
ANCORAGEM(k) = ATRITO(k)
ANCORAGEM(Nsec + 1 - k) =
ANCORAGEM(k)
End If
End If
Next k
Else
If A_indesloc < A_delta Then 'Ponto está
além do meio do vão
Delta_P = (2 * (A_delta - A_indesloc))
/ L
For k = 0 To num_sec
ANCORAGEM(k) = ATRITO(k) -
((2 * (ATRITO(k) - ATRITO(0))) + Delta_P)
If k > 0 Then ANCORAGEM(Nsec
+ 1 - k) = ANCORAGEM(k)
'MsgBox "k: " & k & "P: " &
ANCORAGEM(k)
Next k
Else 'Ponto está antes do meio do vão e
entre duas seções

```

```

If i = 0 Then 'ponto entre num_sec e
meio do vão
a = (x - x_ant) / (4 *
(ATRITO(num_sec) - ATRITO(0)))
b = x_ant
c = A_indesloc_ant - A_delta
Delta_P = SegundoGrau(a, b, c)
L0 = Delta_P * (x - x_ant) / (2 *
(ATRITO(num_sec) - ATRITO(0)))
For k = 0 To num_sec
If k = 0 Then
ANCORAGEM(k) =
ATRITO(k)
Else
ANCORAGEM(k) =
ATRITO(k) - (2 * (ATRITO(k) -
ATRITO(num_sec)) + Delta_P)
ANCORAGEM(Nsec + 1 - k) =
ANCORAGEM(k)
End If
Next k
Else
a = (x - x_ant) / (4 * (ATRITO(i - 1)
- ATRITO(i)))
b = x_ant
c = A_indesloc_ant - A_delta
Delta_P = SegundoGrau(a, b, c)
L0 = Delta_P * (x - x_ant) / (2 *
(ATRITO(i - 1) - ATRITO(i)))
For k = 0 To num_sec
If k = 0 Then
ANCORAGEM(k) =
ATRITO(k)
Else
If k < i Then
ANCORAGEM(k) =
ATRITO(k) - (2 * (ATRITO(k) - ATRITO(i - 1)) +
Delta_P)
ANCORAGEM(Nsec + 1 -
k) = ANCORAGEM(k)
Else
ANCORAGEM(k) =
ATRITO(k)
ANCORAGEM(Nsec + 1 -
k) = ANCORAGEM(k)
End If
End If
Next k
End If
End If
End If
Else
'PASSIVA! -----
-----
-----
For i = 2 To Nsec
x = (((i - 1) * L) / (Nsec - 1))
x_ant = (((i - 1) - 1) * L) / (Nsec - 1))
A_indesloc_ant = A_indesloc

```

```

    A_indesloc = A_indesloc + (((ATRITO(i
- 1) - ATRITO(i)) * (x - x_ant) / 2) + ((ATRITO(i -
1) - ATRITO(i)) * x_ant)) * 2
    If A_indesloc >= A_delta Then
        Exit For
    End If
Next i

If A_indesloc = A_delta Then 'Ponto
coincide com uma seção i
    L0 = x
    For k = 0 To Nsec
        If k = 0 Then
            If x > (L / 2) Then
                ANCORAGEM(k) = ATRITO(k)
- (2 * (ATRITO(k) - ATRITO(i)))
            Else
                ANCORAGEM(k) = ATRITO(k)
            End If
        Else
            If k < i Then
                ANCORAGEM(k) = ATRITO(k)
- (2 * (ATRITO(k) - ATRITO(i)))
            Else
                ANCORAGEM(k) = ATRITO(k)
            End If
        End If
    Next k
Else
    If A_indesloc < A_delta Then 'Ponto está
além da viga
        Delta_P = (2 * (A_delta - A_indesloc))
/ L
        For k = 0 To Nsec
            ANCORAGEM(k) = ATRITO(k) -
((2 * (ATRITO(k) - ATRITO(0))) + Delta_P)
        Next k
        Else 'Ponto está entre duas seções
            a = (x - x_ant) / (4 * (ATRITO(i - 1)
- ATRITO(i)))
            b = x_ant
            c = A_indesloc_ant - A_delta
            Delta_P = SegundoGrau(a, b, c)
            L0 = Delta_P * (x - x_ant) / (2 *
(ATRITO(i - 1) - ATRITO(i)))
            For k = 0 To Nsec
                If k = 0 Then
                    If x > (L / 2) Then
                        ANCORAGEM(k) =
ATRITO(k) - (2 * (ATRITO(k) - ATRITO(i - 1)) +
Delta_P) 'AQUI ACHO QUE ESTÁ ERRADO,
NÃO É i-1 E SIM A SEÇÃO ANTES DO MEIO
DO VÃO
                    Else
                        ANCORAGEM(k) =
ATRITO(k)
                    End If
                End If
            Else
                If k < i Then

```

```

                ANCORAGEM(k) =
ATRITO(k) - (2 * (ATRITO(k) - ATRITO(i - 1)) +
Delta_P)
            Else
                ANCORAGEM(k) =
ATRITO(k)
            End If
        End If
    Next k
End If
End If
End If
End Sub

Sub PERDA_ENCURTAMENTO()
    Nsec = Plan3.Range("C13")
    Ac = Plan3.Range("B7")
    Ic = Plan3.Range("D9")
    L = Plan3.Range("B12")
    Ap = Plan3.Range("B36")
    Wi = Plan3.Range("E10")
    Eci = Plan3.Range("E18")
    ultC = Ultima_B(62, 62)
    num_cabos = Plan2.Range("BJ" & ultC)
    Ep = Plan3.Range("E20")

    Cabo_Equivalente
    alfa = Ep / Eci
    ReDim ENCURTAMENTO(0 To Nsec)

    For i = 0 To Nsec
        If i = 0 Then
            x = L / 2
        Else
            x = (((i - 1) * L) / (Nsec - 1))
        End If
        MatrizTensoes (x)
        np = ANCORAGEM(i)
        e = EQUIVALENTE(i, 0)
        Mpp = TENSAO(0, 0) * Wi
        Mpi = np * e
        If (Plan3.Range("C23") = "Pré-Tração" Or
Mpp > Mpi) Then
            Delta_Sigma = alfa * ((np / Ac) + (np * (e ^
2) / Ic))
            ENCURTAMENTO(i) = np - (Delta_Sigma
* Ap / (10 ^ 4))
        Else
            Delta_Sigma = alfa * ((np / Ac) + (((np * (e
^ 2)) - (Mpp * e)) / Ic)) * ((num_cabos - 1) / (2 *
num_cabos))
            ENCURTAMENTO(i) = np - (num_cabos *
Delta_Sigma * Ap / (10 ^ 4))
        End If
    Next i
End Sub

Sub Fluencia_e_Retracao()

```

```

If (Data5.TextBox3 = "" Or Data5.TextBox4 = ""
Or Data5.TextBox5 = "" Or Data5.TextBox6 = ""
Or Data5.TextBox9 = "") Or (Not
(Data5.OptionButton8 Or Data5.OptionButton9 Or
Data5.OptionButton10)) Then
    Data5.TextBox7 = ""
    Data5.TextBox8 = ""
Else
    Ac = Plan3.Range("B7")
    u = Data5.TextBox3
    T = Data5.TextBox4
    Slump = Data5.TextBox5
    T0 = Data5.TextBox6
    Peri = Data5.TextBox9
    If Data5.OptionButton8 Then
        alfa = 1
        s = 0.38
        fc_t = 1.433
    End If
    If Data5.OptionButton9 Then
        alfa = 2
        s = 0.25
        fc_t = 1.267
    End If
    If Data5.OptionButton10 Then
        alfa = 3
        s = 0.2
        fc_t = 1.208
    End If
    'RETRAÇÃO
    If u > 90 Then
        MsgBox "Umidade máxima de 90%!",
vbOKOnly, "Dados incorretos!"
        Data5.TextBox7 = ""
        Data5.TextBox8 = ""
        Exit Sub
    End If
    hfc_cm = (1 + Exp(-7.8 + (0.1 * u))) * 2 *
(Ac * 10000) / Peri
    hfc = hfc_cm / 100
    Ecs_inf = (-6.16 - (u / 484) + ((u ^ 2) / 1590))
* ((33 + (2 * hfc_cm)) / (20.8 + (3 * hfc_cm))) *
(10 ^ -4)
    If Slump > 0 And Slump < 4 Then Ecs_inf =
0.75 * Ecs_inf
    If Slump > 10 And Slump < 15 Then Ecs_inf
= 1.25 * Ecs_inf
    AA = 40
    BB = 116 * (hfc ^ 3) - 282 * (hfc ^ 2) + 220
* hfc - 4.8
    CC = 2.5 * (hfc ^ 3) - 8.8 * hfc + 40.7
    DD = -75 * (hfc ^ 3) + 585 * (hfc ^ 2) + 496
* hfc - 6.8
    EE = -169 * (hfc ^ 4) + 88 * (hfc ^ 3) + 584 *
(hfc ^ 2) - 39 * hfc + 0.8
    t0_fic = (T + 10) * T0 / 30
    t_fic = (T + 10) * 10000 / 30
    t0_fic_porcent = t0_fic / 100
    t_fic_porcent = t_fic / 100

```

```

    Bs_t = ((t_fic_porcent ^ 3) + AA *
(t_fic_porcent ^ 2) + BB * t_fic_porcent) /
((t_fic_porcent ^ 3) + CC * (t_fic_porcent ^ 2) +
DD * t_fic_porcent + EE)
    Bs_t0 = ((t0_fic_porcent ^ 3) + AA *
(t0_fic_porcent ^ 2) + BB * t0_fic_porcent) /
((t0_fic_porcent ^ 3) + CC * (t0_fic_porcent ^ 2) +
DD * t0_fic_porcent + EE)
    Ecs = Ecs_inf * (Bs_t - Bs_t0)

'FLUÊNCIA
t0_fic = alfa * (T + 10) * T0 / 30
t_fic = alfa * (T + 10) * 10000 / 30
fc_t0 = Exp(s * (1 - ((28 / T0) ^ (1 / 2))))
'rápida
fi_a = 0.8 * (1 - (fc_t0 / fc_t))
'lenta irreversível
fi_1c = 4.45 - (0.035 * u)
If Slump > 0 And Slump < 4 Then fi_1c =
0.75 * fi_1c
If Slump > 10 And Slump < 15 Then fi_1c =
1.25 * fi_1c
fi_2c = (42 + hfc_cm) / (20 + hfc_cm)
AA = 42 * (hfc ^ 3) - 350 * (hfc ^ 2) + 588 *
hfc + 113
BB = 768 * (hfc ^ 3) - 3060 * (hfc ^ 2) +
3234 * hfc - 23
CC = -200 * (hfc ^ 3) + 13 * (hfc ^ 2) + 1090
* hfc + 183
DD = 7579 * (hfc ^ 3) - 31916 * (hfc ^ 2) +
35343 * hfc + 1931
Bf_t0 = ((t0_fic ^ 2) + AA * t0_fic + BB) /
((t0_fic ^ 2) + CC * t0_fic + DD)
Bf_t = ((t_fic ^ 2) + AA * t_fic + BB) / ((t_fic
^ 2) + CC * t_fic + DD)
fi_f = fi_1c * fi_2c * (Bf_t - Bf_t0)
'lenta reversível
fi_d = 0.4 * (t_fic - t0_fic + 20) / (t_fic - t0_fic
+ 70)
fi = fi_a + fi_f + fi_d
Data5.TextBox7 = Format(fi, "0.0000")
Data5.TextBox8 = Format(Ecs, "0.00E+00")
End If
End Sub

Sub PERDAS_PROGRESSIVAS()
Nsec = Plan3.Range("C13")
Ac = Plan3.Range("B7")
Ic = Plan3.Range("D9")
L = Plan3.Range("B12")
Ap = Plan3.Range("B36")
Fptk = Plan2.Range("BK4")
Fio_Cord_Bar = Plan2.Range("BK7")
Nivel_relax = Plan2.Range("BK2")
fi = Plan3.Range("B48")
Ecs = Plan3.Range("B49")
fck = Plan3.Range("B16")
ult = Ultima_B(18, 18)
Wi = Plan3.Range("E10")
Ep = Plan3.Range("E20")

```

```

ReDim PROGRESSIVAS(0 To Nsec)
For i = 0 To Nsec
'Relaxação
Sigma_pi = 10 * ENCURTAMENTO(i) / Ap
'MPa
razao = Sigma_pi / Fptk
If razao <= 0.5 Then linha = 11
If razao > 0.5 And razao <= 0.6 Then linha =
12
If razao > 0.6 And razao <= 0.7 Then linha =
13
If razao > 0.7 And razao <= 0.8 Then linha =
14
If razao > 0.8 Then linha = 14
If Fio_Cord_Bar = "C" And Nivel_relax =
"RN" Then coluna = 71
If Fio_Cord_Bar = "C" And Nivel_relax =
"RB" Then coluna = 72
If Fio_Cord_Bar = "F" And Nivel_relax =
"RN" Then coluna = 73
If Fio_Cord_Bar = "F" And Nivel_relax =
"RB" Then coluna = 74
If Fio_Cord_Bar = "B" Then coluna = 75
If razao >= 0.8 Or razao <= 0.5 Then
Psi = Plan2.Cells(linha, coluna) * 2.5
Else
Psi = 2.5 * (Plan2.Cells(linha - 1, coluna) +
(((razao - Plan2.Cells(linha - 1, 70)) /
(Plan2.Cells(linha, 70) - Plan2.Cells(linha - 1, 70)))
* (Plan2.Cells(linha, coluna) - Plan2.Cells(linha - 1,
coluna))))
End If
'PROCESSO SIMPLIFICADO
khi = -Log(1 - Psi)
khi_c = 1 + (0.5 * fi)
khi_p = 1 + khi
eta = 1 + ((EQUIVALENTE(i, 0) ^ 2) * Ac /
Ic)
rho = (Ap / 10000) / Ac
alfa = Ep / Plan3.Range("E18")
sigma_c_p0 = (ENCURTAMENTO(i) * (-1) /
Ac) + (ENCURTAMENTO(i) * (-1) *
(EQUIVALENTE(i, 0) ^ 2) / Ic) 'KN/m²
If i = 0 Then
x = L / 2
Else
x = ((i - 1) * L) / (Nsec - 1)
End If
MatrizTensoes (x)
permanente = TENSAO(0, 0) * Wi
For k = 2 To ult
categoria = Plan2.Range("W" & k)
If categoria = "Perm" Then permanente =
permanente + (TENSAO(k - 1, 0) * Wi)
If categoria = "Var" Then
Psi2 = Plan2.Range("Z" & k)
permanente = permanente + (TENSAO(k
- 1, 0) * Wi * Psi2) 'KN/m²
End If
Next k

```

```

sigma_c_g = permanente *
EQUIVALENTE(i, 0) / Ic
sigma_c_p0g = (sigma_c_p0 + sigma_c_g) /
1000 'MPa
denominador = (khi_p + (khi_c * alfa * eta *
rho))
If denominador < 1 Then denominador = 1
PERDA = (((Ecs * Ep) + (alfa * sigma_c_p0g
* fi) - (Sigma_pi * khi)) / denominador) * Ap / 10
PROGRESSIVAS(i) = ENCURTAMENTO(i)
+ PERDA
Next i
End Sub

```

```

Sub ELS_Verificacao()
Nsec = Plan3.Range("C13")
L = Plan3.Range("B12")
Pi = Plan3.Range("B37")
NivelProt = Plan3.Range("C24")
Ac = Plan3.Range("B7")
Wi = Plan3.Range("E10")
Ws = Plan3.Range("E11")
fck = Plan3.Range("B16") 'MPa
fator = Plan2.Range("O17")
fctf = fator * 0.7 * 0.3 * (fck ^ (2 / 3)) * 1000
'KN/m²

```

```

ActiveSheet.Range("A1:G4").Select
With Selection
.HorizontalAlignment = xlCenter
.VerticalAlignment = xlCenter
.MergeCells = True
End With
With Selection.Font
.Size = 24
.Bold = True
End With
ActiveSheet.Range("A1") = "ELS -
VERIFICAÇÃO"
k = 0
For i = 0 To Nsec

ActiveSheet.Range("A" & (i + k + 22), "G" &
(i + k + 22)).Merge
ActiveSheet.Range("A" & (i + k +
22)).Interior.Color = RGB(220, 230, 241)
If i = 0 Then
ActiveSheet.Range("A" & (i + k + 22)) =
"MEIO DO VÃO"
x = L / 2
Else
ActiveSheet.Range("A" & (i + k + 22)) =
"SEÇÃO " & i
x = ((i - 1) * L) / (Nsec - 1)
End If
ActiveSheet.Range("A" & (i + k + 23)) =
"Posição:"

```



```

ActiveSheet.Range("B" & (i + k + 23)) =
Format(x, "Standard")
ActiveSheet.Range("C" & (i + k + 23)) = "m"
ActiveSheet.Range("A" & (i + k + 24)) =
"Excentricidade:"
ActiveSheet.Range("B" & (i + k + 24)) =
Format(EQUIVALENTE(i, 0), "Standard")
ActiveSheet.Range("C" & (i + k + 24)) = "m"
ActiveSheet.Range("A" & (i + k + 25)) =
"Força inicial de protensão:"
ActiveSheet.Range("B" & (i + k + 25)) =
Format(Pi, "Standard")
ActiveSheet.Range("C" & (i + k + 25)) =
"KN"
ActiveSheet.Range("A" & (i + k + 26)) =
"Perda por atrito:"
ActiveSheet.Range("B" & (i + k + 26)) =
Format((Pi - ATRITO(i)), "Standard")
ActiveSheet.Range("C" & (i + k + 26)) =
"KN"
ActiveSheet.Range("A" & (i + k + 27)) =
"Perda por ancoragem:"
ActiveSheet.Range("B" & (i + k + 27)) =
Format((ATRITO(i) - ANCORAGEM(i)),
"Standard")
ActiveSheet.Range("C" & (i + k + 27)) =
"KN"
ActiveSheet.Range("A" & (i + k + 28)) =
"Perda por encurtamento:"
ActiveSheet.Range("B" & (i + k + 28)) =
Format((ANCORAGEM(i) -
ENCURTAMENTO(i)), "Standard")
ActiveSheet.Range("C" & (i + k + 28)) =
"KN"
ActiveSheet.Range("A" & (i + k + 29)) =
"Perdas progressivas:"
ActiveSheet.Range("B" & (i + k + 29)) =
Format((ENCURTAMENTO(i) -
PROGRESSIVAS(i)), "Standard")
ActiveSheet.Range("C" & (i + k + 29)) =
"KN"
ActiveSheet.Range("A" & (i + k + 30)) =
"Força final de protensão:"
ActiveSheet.Range("B" & (i + k + 30)) =
Format(PROGRESSIVAS(i), "Standard")
ActiveSheet.Range("C" & (i + k + 30)) =
"KN"
ActiveSheet.Range("A" & (i + k + 31)) =
"Porcentagem total de perdas:"
ActiveSheet.Range("B" & (i + k + 31)) =
Format(((Pi - PROGRESSIVAS(i)) * 100 / Pi),
"Standard")
ActiveSheet.Range("C" & (i + k + 31)) = "%"

MatrizTensoes (x)

If NivelProt = "Nível 3 - Protensão Completa"
Then
CF = Comb_Frequente_Inferior()
CR = Comb_Rara_Inferior()

```

```

CRS = Comb_Rara_Superior()

T_CF = CF - (PROGRESSIVAS(i) / Ac) -
(PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Wi)
T_CR = CR - (PROGRESSIVAS(i) / Ac) -
(PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Wi)
T_CRS = CRS - (PROGRESSIVAS(i) / Ac)
- (PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Ws)

ActiveSheet.Range("A" & (i + k + 32)) =
"Tensão na borda inferior (CF):"
ActiveSheet.Range("B" & (i + k + 32)) =
Format(T_CF, "Standard")
ActiveSheet.Range("C" & (i + k + 32)) =
"KN/m²"
ActiveSheet.Range("A" & (i + k + 33)) =
"Tensão na borda inferior (CR):"
ActiveSheet.Range("B" & (i + k + 33)) =
Format(T_CR, "Standard")
ActiveSheet.Range("C" & (i + k + 33)) =
"KN/m²"
ActiveSheet.Range("A" & (i + k + 34)) =
"Tensão na borda superior (CRS):"
ActiveSheet.Range("B" & (i + k + 34)) =
Format(T_CRS, "Standard")
ActiveSheet.Range("C" & (i + k + 34)) =
"KN/m²"

If T_CF <= 0 Then
ActiveSheet.Range("D" & (i + k + 32)) =
"ELS-D OK!"
ActiveSheet.Range("D" & (i + k +
32)).Interior.Color = RGB(0, 255, 0)
Else
ActiveSheet.Range("D" & (i + k + 32)) =
"ELS-D NÃO OK!"
ActiveSheet.Range("D" & (i + k +
32)).Interior.Color = RGB(255, 0, 0)
MsgBox "ELS-D: Limite atingido na
seção " & i
End If
If T_CR <= fctf Then
ActiveSheet.Range("D" & (i + k + 33)) =
"ELS-F OK!"
ActiveSheet.Range("D" & (i + k +
33)).Interior.Color = RGB(0, 255, 0)
Else
ActiveSheet.Range("D" & (i + k + 33)) =
"ELS-F NÃO OK!"
ActiveSheet.Range("D" & (i + k +
33)).Interior.Color = RGB(255, 0, 0)
MsgBox "ELS-F: Limite atingido na
seção " & i
End If
If T_CRS >= (-0.6 * fck * 1000) Then
ActiveSheet.Range("D" & (i + k + 34)) =
"ELS-CE OK!"

```

```

        ActiveSheet.Range("D" & (i + k +
34)).Interior.Color = RGB(0, 255, 0)
    Else
        ActiveSheet.Range("D" & (i + k + 34)) =
"ELS-CE NÃO OK!"
        ActiveSheet.Range("D" & (i + k +
34)).Interior.Color = RGB(255, 0, 0)
        MsgBox "ELS-CE: Limite atingido na
seção " & i
    End If

End If

If NivelProt = "Nível 2 - Protensão Limitada"
Then
    CF = Comb_Frequente_Inferior()
    CQP = Comb_QuasePermanente_Inferior()
    CFS = Comb_Frequente_Superior()

    T_CF = CF - (PROGRESSIVAS(i) / Ac) -
(PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Wi)
    T_CQP = CQP - (PROGRESSIVAS(i) / Ac)
- (PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Wi)
    T_CFS = CFS - (PROGRESSIVAS(i) / Ac)
- (PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Ws)

    ActiveSheet.Range("A" & (i + k + 32)) =
"Tensão na borda inferior (CF):"
    ActiveSheet.Range("B" & (i + k + 32)) =
Format(T_CF, "Standard")
    ActiveSheet.Range("C" & (i + k + 32)) =
"KN/m²"
    ActiveSheet.Range("A" & (i + k + 33)) =
"Tensão na borda inferior (CQP):"
    ActiveSheet.Range("B" & (i + k + 33)) =
Format(T_CQP, "Standard")
    ActiveSheet.Range("C" & (i + k + 33)) =
"KN/m²"
    ActiveSheet.Range("A" & (i + k + 34)) =
"Tensão na borda superior (CFS):"
    ActiveSheet.Range("B" & (i + k + 34)) =
Format(T_CFS, "Standard")
    ActiveSheet.Range("C" & (i + k + 34)) =
"KN/m²"

    If T_CQP <= 0 Then
        ActiveSheet.Range("D" & (i + k + 33)) =
"ELS-D OK!"
        ActiveSheet.Range("D" & (i + k +
33)).Interior.Color = RGB(0, 255, 0)
    Else
        ActiveSheet.Range("D" & (i + k + 33)) =
"ELS-D NÃO OK!"
        ActiveSheet.Range("D" & (i + k +
33)).Interior.Color = RGB(255, 0, 0)
        MsgBox "ELS-D: Limite atingido na
seção " & i
    End If
End If

```

```

End If
If T_CF <= fctf Then
    ActiveSheet.Range("D" & (i + k + 32)) =
"ELS-F OK!"
    ActiveSheet.Range("D" & (i + k +
32)).Interior.Color = RGB(0, 255, 0)
    Else
        ActiveSheet.Range("D" & (i + k + 32)) =
"ELS-F NÃO OK!"
        ActiveSheet.Range("D" & (i + k +
32)).Interior.Color = RGB(255, 0, 0)
        MsgBox "ELS-F: Limite atingido na
seção " & i
    End If
    If T_CFS >= (-0.6 * fck * 1000) Then
        ActiveSheet.Range("D" & (i + k + 34)) =
"ELS-CE OK!"
        ActiveSheet.Range("D" & (i + k +
34)).Interior.Color = RGB(0, 255, 0)
    Else
        ActiveSheet.Range("D" & (i + k + 34)) =
"ELS-CE NÃO OK!"
        ActiveSheet.Range("D" & (i + k +
34)).Interior.Color = RGB(255, 0, 0)
        MsgBox "ELS-CE: Limite atingido na
seção " & i
    End If
End If

End If

If NivelProt = "Nível 1 - Protensão
Parcial" Then
    CF = Comb_Frequente_Inferior()
    CFS = Comb_Frequente_Superior()

    T_CF = CF - (PROGRESSIVAS(i) / Ac) -
(PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Wi)
    T_CFS = CFS - (PROGRESSIVAS(i) / Ac)
- (PROGRESSIVAS(i) * EQUIVALENTE(i, 0) /
Ws)
    ActiveSheet.Range("A" & (i + k + 32)) =
"Tensão na borda inferior (CF):"
    ActiveSheet.Range("B" & (i + k + 32)) =
Format(T_CF, "Standard")
    ActiveSheet.Range("C" & (i + k + 32)) =
"KN/m²"
    ActiveSheet.Range("A" & (i + k + 33)) =
"Tensão na borda superior (CFS):"
    ActiveSheet.Range("B" & (i + k + 33)) =
Format(T_CFS, "Standard")
    ActiveSheet.Range("C" & (i + k + 33)) =
"KN/m²"

    If T_CF <= fctf Then
        ActiveSheet.Range("D" & (i + k + 32)) =
"ELS-F OK!"
        ActiveSheet.Range("D" & (i + k +
32)).Interior.Color = RGB(0, 255, 0)
    Else

```

```

ActiveSheet.Range("D" & (i + k + 32)) =
"VER FISS."
ActiveSheet.Range("D" & (i + k +
32)).Interior.Color = RGB(255, 0, 0)
MsgBox "ELS-F: Limite atingido na
seção " & i
End If
If T_CFS >= (-0.6 * fck * 1000) Then
ActiveSheet.Range("D" & (i + k + 33)) =
"ELS-CE OK!"
ActiveSheet.Range("D" & (i + k +
33)).Interior.Color = RGB(0, 255, 0)
Else
ActiveSheet.Range("D" & (i + k + 33)) =
"ELS-CE NÃO OK!"
ActiveSheet.Range("D" & (i + k +
33)).Interior.Color = RGB(255, 0, 0)
MsgBox "ELS-CE: Limite atingido na
seção " & i
End If
End If

k = k + 13
Next i
ActiveSheet.Columns.AutoFit
ActiveSheet.Range("B23", "B" & (((Nsec + 1) *
14) + 21)).HorizontalAlignment = xlCenter

'GRÁFICO PERDAS
Plan2.Range("CG3", "CM" & (Ultima_B(85, 91)
+ 1)).Clear
For i = 0 To Nsec
If i = 0 Then
x = L / 2
Else
x = ((i - 1) * L) / (Nsec - 1)
End If
Plan2.Range("CG" & (i + 3)) = i
Plan2.Range("CH" & (i + 3)) = x
Plan2.Range("CI" & (i + 3)) = Pi
Plan2.Range("CJ" & (i + 3)) = ATRITO(i)
Plan2.Range("CK" & (i + 3)) =
ANCORAGEM(i)
Plan2.Range("CL" & (i + 3)) =
ENCURTAMENTO(i)
Plan2.Range("CM" & (i + 3)) =
PROGRESSIVAS(i)
Next i
ult = Ultima_B(85, 85)
MaxScaleY =
(Application.WorksheetFunction.RoundUp((Plan2.
Range("CI3") / 100), 0) + 1) * 100
MinScaleY =
(Application.WorksheetFunction.RoundDown((Pla
n2.Range("CM4") / 100), 0) - 1) * 100
Sheets("ELS").Select
ActiveSheet.Shapes.AddChart.Select
With ActiveChart
.ChartType = xlXYScatterLines
.SeriesCollection.NewSeries

```

```

.SeriesCollection(1).Name = ""antes das
perdas""
.SeriesCollection(1).XValues =
"=BIBLIOTECA!$CH$4:$CH$" & ult
.SeriesCollection(1).Values =
"=BIBLIOTECA!$CI$4:$CI$" & ult
.SeriesCollection.NewSeries
.SeriesCollection(2).Name = ""após
atrito""
.SeriesCollection(2).XValues =
"=BIBLIOTECA!$CH$4:$CH$" & ult
.SeriesCollection(2).Values =
"=BIBLIOTECA!$CJ$4:$CJ$" & ult
.SeriesCollection.NewSeries
.SeriesCollection(3).Name = ""após
encunhamento""
.SeriesCollection(3).XValues =
"=BIBLIOTECA!$CH$4:$CH$" & ult
.SeriesCollection(3).Values =
"=BIBLIOTECA!$CK$4:$CK$" & ult
.SeriesCollection.NewSeries
.SeriesCollection(4).Name = ""após
encurtamento""
.SeriesCollection(4).XValues =
"=BIBLIOTECA!$CH$4:$CH$" & ult
.SeriesCollection(4).Values =
"=BIBLIOTECA!$CL$4:$CL$" & ult
.SeriesCollection.NewSeries
.SeriesCollection(5).Name = ""após
progressivas""
.SeriesCollection(5).XValues =
"=BIBLIOTECA!$CH$4:$CH$" & ult
.SeriesCollection(5).Values =
"=BIBLIOTECA!$CM$4:$CM$" & ult
.HasTitle = True
.ChartTitle.Characters.Text = "FORÇA AO
LONGO DO CABO"
.Axes(xlCategory, xlPrimary).HasTitle = True
.Axes(xlCategory,
xlPrimary).AxisTitle.Characters.Text =
"Comprimento (m)"
.Axes(xlValue, xlPrimary).HasTitle = True
.Axes(xlValue,
xlPrimary).AxisTitle.Characters.Text = "Força
(kN)"
.Axes(xlValue).MaximumScale = MaxScaleY
.Axes(xlValue).MinimumScale = MinScaleY
.Axes(xlCategory).MaximumScale = L
.Axes(xlCategory).MinimumScale = 0
.Axes(xlCategory,
xlPrimary).HasMajorGridlines = True
.Axes(xlValue, xlPrimary).HasMajorGridlines
= True
.Axes(xlValue, xlPrimary).HasMinorGridlines
= True

.Axes(xlCategory).Format.Line.ForeColor.RGB =
RGB(0, 0, 0)
.Axes(xlValue).Format.Line.ForeColor.RGB =
RGB(0, 0, 0)

```

```
.Axes(xlCategory).MajorUnit = L / (Nsec - 1)
.Axes(xlCategory).TickLabels.Orientation =
45
.SeriesCollection(6).Delete
.SeriesCollection(6).Delete
.SeriesCollection(6).Delete
.SeriesCollection(6).Delete
.ChartTitle.Text = "Força de protensão ao
longo do cabo"
.Parent.Top = 75
.Parent.Left = 25
.ChartArea.Width = Nsec * 350 / 11
End With
End Sub
```