

### 3

## O Programa Computacional ANLOG – Versão 3D

### 3.1.

#### Apresentação

O programa computacional ANLOG (Análise não linear de obras geotécnicas) é escrito em linguagem de programação FORTRAN 90. Sua primeira versão foi desenvolvida por Zornberg (1989) tendo sido utilizada na análise de problemas mecânicos de equilíbrio estático, envolvendo a simulação de aterros e escavações, sob condição de deformação plana e axissimétrica e levando em conta comportamento tensão-deformação elastoplásticos.

Uma nova versão do programa foi desenvolvida por Nogueira (1998) considerando o acoplamento entre fluxo e deformação. Dando continuidade aos trabalhos desenvolvidos por Nogueira (1998), outras versões deste código foram desenvolvidas. Machado Jr. (2000) desenvolveu uma versão (GEOFLUX) para análise de problemas de fluxo em meio poroso não saturado. Pereira (2003) incorporou os elementos de reforço que possibilitou a análise de problemas mecânicos de equilíbrio estático de estruturas de solos reforçados (ANLOG v.2003). Pinto (2004) incorporou à formulação acoplada o efeito da variação do nível d'água (ANLOG v.2004). Silva (2005) introduziu os elementos tridimensionais. Oliveira (2006) incorporou ao código os elementos de reforço e interface axissimétricos e os modelos Mohr-Coulomb modificado para o solo e Coulomb para a interface com comportamento elástico-perfeitamente plástico (ANLOG v.2006).

Recentemente, Yang (2009) incorporou a versão do modelo Lade-Kim proposta por Lade e Jacobsen (2002), assim como, sugeriu uma nova modificação para a função de amolecimento a fim de suavizar a curva tensão-deformação na condição pós-pico e ainda levar em conta o efeito da tensão de confinamento na resistência residual. Essa nova versão (ANLOG v. 2009) foi verificada e calibrada a partir dos resultados de ensaio em centrífuga desenvolvidos por Arriaga (2001).

O ANLOG roda numa plataforma Windows (Figure 3.1) e o programa MTOOL (Figure 3.2) desenvolvido pelo grupo de tecnologia em computação gráfica da PUC-Rio (TecGraf<sup>®</sup>) é usado como pré e pós processadores gráficos das análises planas, e o pré e pós processador GID para as análises 3D.

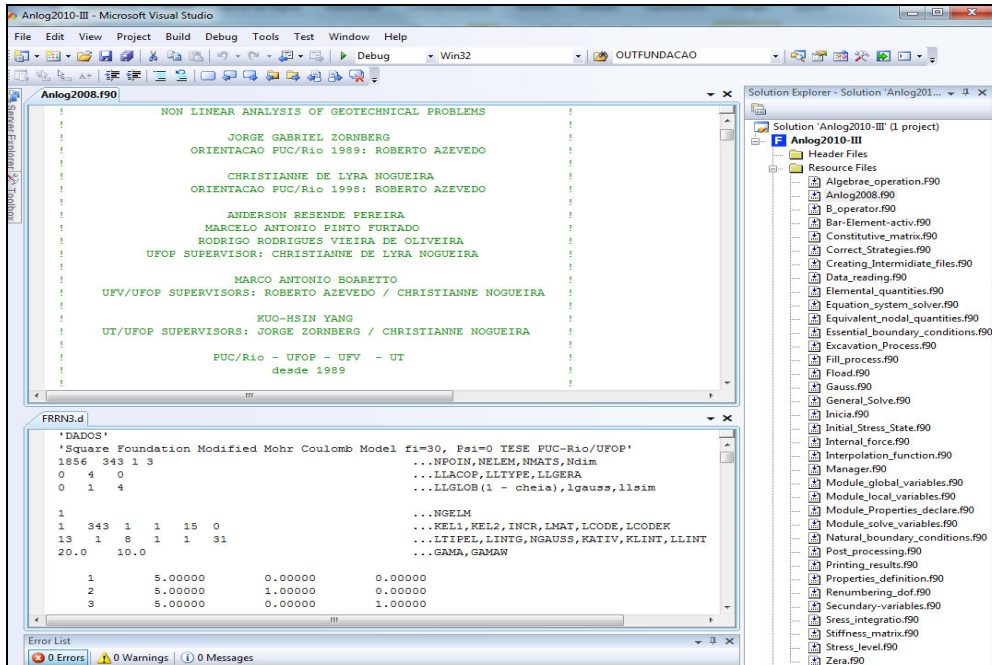


Figura 3.1 – Ambiente de trabalho do FORTRAN

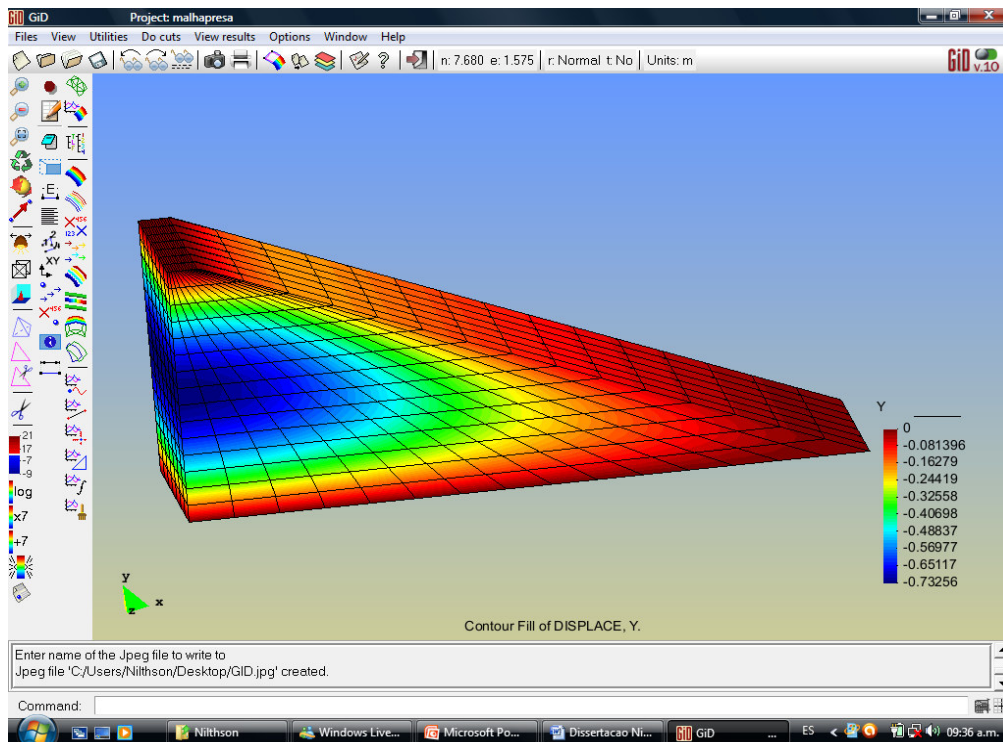


Figura 3.2 – Pre e post Processador GID V10

Em todas as versões, a estrutura em macro-comando é adotada. Esta estrutura permite a simulação de processos construtivos tais como aterros e escavações. Este programa foi utilizado para a análise de diferentes casos históricos (Zornberg, 1989; Nogueira, 1992; Azevedo *et al.*, 1994; Azevedo e Azevedo, 1994; Nogueira et al, 2010) e uma boa concordância tem sido observada entre os resultados numéricos e de campo.

Um macro-comando é uma palavra chave utilizada para controlar a execução de blocos de rotinas que devem ser acionadas para realização de uma tarefa específica. O usuário deverá fornecer a sequência de macro-comandos e todos os dados relacionados a eles. Esta sequência de macro-comandos define o fluxo de informação que deverá ser usado na solução de um problema específico. Alguns exemplos de macro-comandos são apresentados na Figura 2.3.

O macro-comando DADOS ativa um bloco de rotinas as quais são responsáveis pela leitura dos dados geométricos (coordenadas e conectividades), materiais (modelos constitutivos e parâmetros) e condições de contorno (essencial).

Os macro-comandos CPOIN, CEDGE e CGRAV ativam um bloco de rotinas que fornecem o carregamento nodal devido às forças pontuais, de superfície e de volume (ou corpo), respectivamente.

O macro-comando SOLVE é usado para se obter as matrizes características globais e resolver o sistema de equação algébrico característico do problema. As variáveis secundárias (deformação, tensão, gradiente hidráulico, velocidade de fluxo, etc..) são também obtidas através deste macro-comando.

Os macro-comandos TINK0 e TINIS são usados para definir um estado de tensão inicial considerando um estado de deformação inicial nulo. O macro-comando TINK0 é usado para se obter um estado de tensão geostático e o TINIS para um estado de tensão isotrópico.

Existem vários macro-comandos disponíveis no ANLOG além dos citados anteriormente, tais como o macro comando CPRES para definição de condições de contorno prescritas não nulas em deslocamento; ATERR para simulação da construção de aterros; POSR para criação de um arquivo de saída gráfica compatível com o MTOOL, e GID V10 etc.

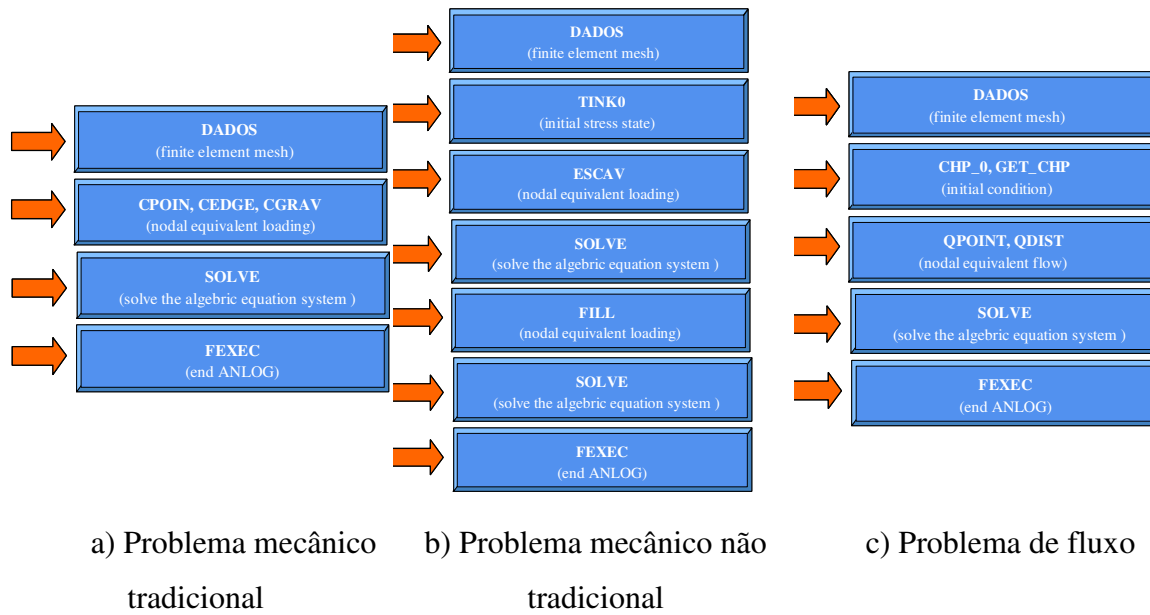


Figura 3.3 – Exemplo de sequencias de macro-comandos (Nogueira 1998)

### 3.2. Elementos finitos sólidos

Para análise tridimensional, o elemento cúbico (ou sólido) quadrático (C20) foi adotado. Este elemento é apresentado na Figura 3.4 no sistema de coordenada natural ( $\xi, \eta, \zeta$ ).

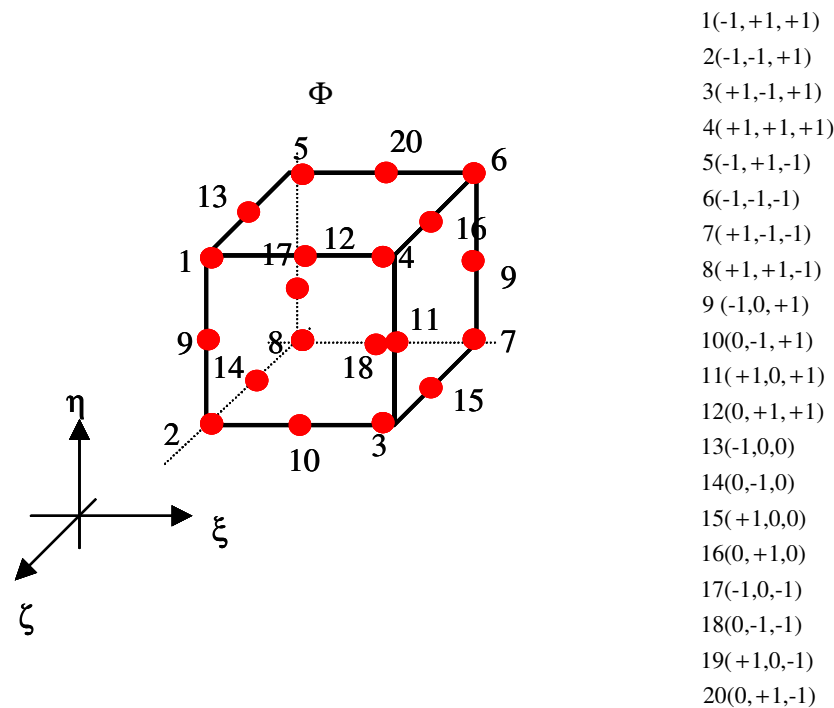


Figura 3.4 – Elementos finitos sólidos C20

As funções de forma  $N_i$  para este elemento são definidas no sistema de coordenada natural como:

$$N_1(\xi, \eta) = 0.125(1-\xi)(1+\eta)(1+\zeta)(-\xi+\eta+\zeta-2) \quad (3.1)$$

$$N_2(\xi, \eta) = 0.125(1-\xi)(1-\eta)(1+\zeta)(-\xi-\eta+\zeta-2) \quad (3.2)$$

$$N_3(\xi, \eta) = 0.125(1+\xi)(1-\eta)(1+\zeta)(+\xi-\eta+\zeta-2) \quad (3.3)$$

$$N_4(\xi, \eta) = 0.125(1+\xi)(1+\eta)(1+\zeta)(+\xi+\eta+\zeta-2) \quad (3.4)$$

$$N_5(\xi, \eta) = 0.125(1-\xi)(1+\eta)(1-\zeta)(-\xi+\eta-\zeta-2) \quad (3.5)$$

$$N_6(\xi, \eta) = 0.125(1-\xi)(1-\eta)(1-\zeta)(-\xi-\eta-\zeta-2) \quad (3.6)$$

$$N_7(\xi, \eta) = 0.125(1+\xi)(1-\eta)(1-\zeta)(+\xi-\eta-\zeta-2) \quad (3.7)$$

$$N_8(\xi, \eta) = 0.125(1+\xi)(1+\eta)(1-\zeta)(+\xi+\eta-\zeta-2) \quad (3.8)$$

$$N_9(\xi, \eta) = 0.25(1-\xi)(1-\eta^2)(1+\zeta) \quad (3.9)$$

$$N_{10}(\xi, \eta) = 0.25(1-\xi^2)(1-\eta)(1+\zeta) \quad (3.10)$$

$$N_{11}(\xi, \eta) = 0.25(1+\xi)(1-\eta^2)(1+\zeta) \quad (3.11)$$

$$N_{12}(\xi, \eta) = 0.25(1-\xi^2)(1+\eta)(1+\zeta) \quad (3.12)$$

$$N_{13}(\xi, \eta) = 0.25(1-\xi)(1+\eta)(1-\zeta^2) \quad (3.13)$$

$$N_{14}(\xi, \eta) = 0.25(1-\xi)(1-\eta)(1-\zeta^2) \quad (3.14)$$

$$N_{15}(\xi, \eta) = 0.25(1+\xi)(1-\eta)(1-\zeta^2) \quad (3.15)$$

$$N_{16}(\xi, \eta) = 0.25(1+\xi)(1+\eta)(1-\zeta^2) \quad (3.16)$$

$$N_{17}(\xi, \eta) = 0.25(1-\xi)(1-\eta^2)(1-\zeta) \quad (3.17)$$

$$N_{18}(\xi, \eta) = 0.25(1-\xi^2)(1-\eta)(1-\zeta) \quad (3.18)$$

$$N_{19}(\xi, \eta) = 0.25(1+\xi)(1-\eta^2)(1-\zeta) \quad (3.19)$$

$$N_{20}(\xi, \eta) = 0.25(1-\xi^2)(1+\eta)(1-\zeta) \quad (3.20)$$

### 3.3. Aproximação da geometria

As coordenadas de um determinado ponto são escritas em função das coordenadas dos pontos nodais através das funções de interpolação quadráticas.

$$\mathbf{x} = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} = \mathbf{N}\hat{\mathbf{x}} \quad (3.21)$$

onde  $\mathbf{x}^T = [x_1 \ y_1 \ z_1 \ \dots \ x_{20} \ y_{20} \ z_{20}]$  é o vetor das coordenadas locais dos pontos nodais.  $\mathbf{N}$  é a matriz das funções de forma definida como:

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 & 0 & N_{20} & 0 & 0 \\ 0 & N_1 & 0 & \dots & 0 & N_{20} & 0 \\ 0 & 0 & N_1 & 0 & 0 & 0 & N_{20} \end{bmatrix} \quad (3.22)$$

em função das funções de forma  $N_i$  (Equação 3.1 a 3.20).

A matriz jacobiana  $\mathbf{J}$  é utilizada na transformação de sistemas de coordenadas local-natural ( $xyz$  e  $\xi\eta\zeta$ ) e é definida como:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix} \quad (3.23)$$

em que

$$\begin{Bmatrix} \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \xi} \end{Bmatrix} = \begin{bmatrix} \frac{\partial N_1}{\partial \xi} & 0 & 0 & \dots & \frac{\partial N_{20}}{\partial \xi} & 0 & 0 \\ 0 & \frac{\partial N_1}{\partial \xi} & 0 & \dots & 0 & \frac{\partial N_{20}}{\partial \xi} & 0 \\ 0 & 0 & \frac{\partial N_1}{\partial \xi} & \dots & 0 & 0 & \frac{\partial N_{20}}{\partial \xi} \end{bmatrix} \begin{Bmatrix} x_1 \\ y_1 \\ z_1 \\ \vdots \\ x_{20} \\ y_{20} \\ z_{20} \end{Bmatrix} \quad (3.24a)$$

$$\left\{ \begin{array}{l} \partial x / \partial \eta \\ \partial y / \partial \eta \\ \partial z / \partial \eta \end{array} \right\} = \left[ \begin{array}{ccccccc} \partial N_1 / \partial \eta & 0 & 0 & \partial N_{20} / \partial \eta & 0 & 0 & \\ 0 & \partial N_1 / \partial \eta & 0 & \dots & 0 & \partial N_{20} / \partial \eta & 0 \\ 0 & 0 & \partial N_1 / \partial \eta & & 0 & 0 & \partial N_{20} / \partial \eta \end{array} \right] \left\{ \begin{array}{l} x_1 \\ y_1 \\ z_1 \\ \vdots \\ x_{20} \\ y_{20} \\ z_{20} \end{array} \right\} \quad (3.24b)$$

$$\left\{ \begin{array}{l} \partial x / \partial \zeta \\ \partial y / \partial \zeta \\ \partial z / \partial \zeta \end{array} \right\} = \left[ \begin{array}{ccccccc} \partial N_1 / \partial \zeta & 0 & 0 & \partial N_{20} / \partial \zeta & 0 & 0 & \\ 0 & \partial N_1 / \partial \zeta & 0 & \dots & 0 & \partial N_{20} / \partial \zeta & 0 \\ 0 & 0 & \partial N_1 / \partial \zeta & & 0 & 0 & \partial N_{20} / \partial \zeta \end{array} \right] \left\{ \begin{array}{l} x_1 \\ y_1 \\ z_1 \\ \vdots \\ x_{20} \\ y_{20} \\ z_{20} \end{array} \right\} \quad (3.24c)$$

O volume elementar  $dV = dx dy dz$  pode ser escrito em coordenadas naturais como

$$dV = \det \mathbf{J}(\xi, \zeta, \eta) d\xi d\eta d\zeta \quad (3.25)$$

em que

$$\det \mathbf{J} = \left[ \frac{\partial z}{\partial \xi} \left( \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \zeta} - \frac{\partial x}{\partial \zeta} \frac{\partial y}{\partial \eta} \right) + \frac{\partial z}{\partial \eta} \left( \frac{\partial x}{\partial \zeta} \frac{\partial y}{\partial \xi} - \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \zeta} \right) + \frac{\partial z}{\partial \zeta} \left( \frac{\partial x}{\partial \xi} \frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta} \frac{\partial y}{\partial \xi} \right) \right] \quad (3.26)$$

### 3.4. Operador diferencial

A matriz  $\mathbf{B}$  que relaciona as componentes de deformação com as componentes de deslocamentos nodais é dada por:

$$\mathbf{B} = [\mathbf{B}_1 \quad \dots \quad \mathbf{B}_{20}] \quad (3.27)$$

em

$$\mathbf{B}_i = \nabla \mathbf{N}_i = \begin{bmatrix} \partial \mathbf{N}_i / \partial x & 0 & 0 \\ 0 & \partial \mathbf{N}_i / \partial y & 0 \\ 0 & 0 & \partial \mathbf{N}_i / \partial z \\ \partial \mathbf{N}_i / \partial y & \partial \mathbf{N}_i / \partial x & 0 \\ 0 & \partial \mathbf{N}_i / \partial z & \partial \mathbf{N}_i / \partial y \\ \partial \mathbf{N}_i / \partial z & 0 & \partial \mathbf{N}_i / \partial x \end{bmatrix} \quad (3.28)$$

Aplicando a regra da cadeia:

$$\begin{pmatrix} \frac{\partial \mathbf{N}_i}{\partial x} \\ \frac{\partial \mathbf{N}_i}{\partial y} \\ \frac{\partial \mathbf{N}_i}{\partial z} \end{pmatrix} = \begin{pmatrix} \frac{\partial \mathbf{N}_i}{\partial \xi} \frac{\partial \xi}{\partial x} + \frac{\partial \mathbf{N}_i}{\partial \eta} \frac{\partial \eta}{\partial x} + \frac{\partial \mathbf{N}_i}{\partial \zeta} \frac{\partial \zeta}{\partial x} \\ \frac{\partial \mathbf{N}_i}{\partial \xi} \frac{\partial \xi}{\partial y} + \frac{\partial \mathbf{N}_i}{\partial \eta} \frac{\partial \eta}{\partial y} + \frac{\partial \mathbf{N}_i}{\partial \zeta} \frac{\partial \zeta}{\partial y} \\ \frac{\partial \mathbf{N}_i}{\partial \xi} \frac{\partial \xi}{\partial z} + \frac{\partial \mathbf{N}_i}{\partial \eta} \frac{\partial \eta}{\partial z} + \frac{\partial \mathbf{N}_i}{\partial \zeta} \frac{\partial \zeta}{\partial z} \end{pmatrix} \quad (3.29)$$

### 3.5. Matriz de rigidez e vetor de força interna

A matriz de rigidez  $\mathbf{K}^e$  que relaciona as componentes de deslocamento com as componentes de força externa, a nível do elemento, é dada por:

$$\mathbf{K}^e = \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \underbrace{\mathbf{B}^T \mathbf{D}_i \mathbf{B}}_{\mathbf{T}^K(\xi, \eta)} \det \mathbf{J} d\xi d\eta d\zeta = \sum_{i=1}^{n_{\text{gauss}}} [w_{\xi_i} w_{\eta_i} w_{\zeta_i} \mathbf{T}^K(\xi_i, \eta_i, \zeta_i)] \quad (3.30)$$

O vetor de força interna  $\mathbf{F}_{\text{int}}^e$  a nível do elemento, é dada por:

$$\mathbf{K}^e = \int_{-1}^{+1} \int_{-1}^{+1} \int_{-1}^{+1} \underbrace{\mathbf{B}^T \boldsymbol{\sigma}}_{\mathbf{T}^\sigma(\xi, \eta, \zeta)} \det \mathbf{J} d\xi d\eta d\zeta = \sum_{i=1}^{n_{\text{gauss}}} [w_{\xi_i} w_{\eta_i} w_{\zeta_i} \mathbf{T}^\sigma(\xi_i, \eta_i, \zeta_i)] \quad (3.31)$$

$\xi_i, \eta_i, \zeta_i$ , são as coordenadas dos pontos de Gauss e  $w_{\xi_i}, w_{\eta_i}, w_{\zeta_i}$  são os pesos dos respectivos pontos de Gauss.

### 3.6. Simulação Numérica via MEF da construção de aterros

Existem na literatura diferentes formas de simulação da construção de aterros via MEF. Este tipo de problema envolve, invariavelmente, três etapas



distintas: a primeira refere-se à definição do vetor de carregamento nodal; a segunda refere-se ao tratamento dado aos elementos ainda não construídos (isto vai interferir na rigidez e na dimensão do sistema de equação a ser resolvido); e, a terceira refere-se à solução do sistema de equação algébrica, a qual fornecerá o efeito da nova camada sobre as camadas pré-existente.

A metodologia adotada no ANLOG para o cálculo do carregamento nodal equivalente é chamada de “ligar a gravidade” e consiste em obter o carregamento nodal equivalente à força de peso próprio da camada que está sendo construída, tal como ilustrado na Figura 3.5.

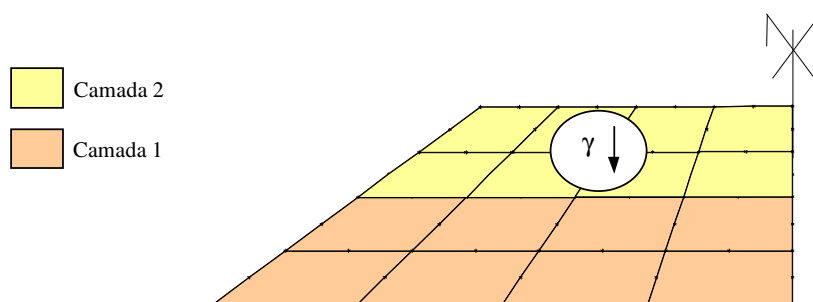


Figura 3.5 – Procedimento de simulação de construção em camadas “Ligar a gravidade”

A desvantagem deste procedimento está relacionada ao fato dos elementos que estão sendo construídos partirem de uma condição de tensão nula, o que normalmente não é aceitável nas equações de modelos constitutivos não-lineares. Para contornar este problema, normalmente estipula-se um “pequeno” estado de tensão inicial para os elementos da camada em construção.

Com relação à segunda etapa mencionada acima, inclusão dos elementos em construção na malha, considera-se uma malha de elementos finitos variável que vai crescendo à medida que novas camadas (elementos) vão sendo construídas na simulação.

Clough e Woodward (1967) mostraram que, ao contrário da escavação, os resultados de uma análise via MEF da construção de um aterro é altamente dependente do número de etapas de construção, mesmo com um material linear elástico. Surge a questão: quantas camadas devem ser usadas para que um resultado razoável seja obtido, sem que seja necessário centenas de camadas para simulação da construção de uma barragem?

Vários estudos têm sido feitos com o intuito de definir-se o número ideal de camadas para simulação numérica de aterros. O que se tem observado é que esse número depende da região de interesse, ou seja, o aterro propriamente dito ou a fundação deste aterro. E, ainda, depende das variáveis que se deseja investigar, ou seja, deslocamentos, tensões ou ambos.

Quando o interesse é analisar apenas o comportamento da fundação do aterro, este pode ser simulado em apenas uma única camada. Porém, quando o objeto de estudo é o aterro propriamente dito, ele deve ser simulado num número maior de camadas, sendo que, quando o interesse está voltado para a avaliação das tensões, o número de camadas pode ser menor que quando o interesse consiste na previsão dos deslocamentos.

Clough e Duncan (1969) sugeriram para uma análise não linear elástica o uso de dez camadas. Este número tem sido um valor típico em diversas análises. Naylor e Mattar (1988) sugerem que este número seja reduzido para 5 ou 6, desde que seja utilizado um fator de redução para a rigidez das camadas em construção. Eles alegam que, como as camadas das simulações numéricas são mais espessas que as camadas reais, então, elas seriam também mais rígidas. Assim sendo, para análises lineares-elásticas, eles sugerem um fator de redução de 4 e, para análises não-lineares, um fator de redução de 3 a 5.